ARRAY COMBINATION FOR PARALLEL IMAGING IN MAGNETIC RESONANCE IMAGING

A Dissertation

by

DAN KENRICK SPENCE

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

May 2006

Major Subject: Electrical Engineering

ARRAY COMBINATION FOR PARALLEL IMAGING IN MAGNETIC RESONANCE IMAGING

A Dissertation

by

DAN KENRICK SPENCE

Submitted to the Office of Graduate Studies of
Texas A&M University
in partial fulfillment of the requirements for the degree of

DOCTOR OF PHILOSOPHY

Approved by:

Chair of Committee,	Steven Wright
Committee Members,	Robert Nevels
	Laszlo Kish
	David Church
Head of Department,	Costas Georghiades

May 2006

Major Subject: Electrical Engineering

ABSTRACT

Array Combination for Parallel Imaging in Magnetic Resonance Imaging. (May 2006)
Dan Kenrick Spence, B.S., Texas A&M University;
M.S., Texas A&M University
Chair of Advisory Committee: Dr. Steve Wright

In Magnetic Resonance Imaging, the time required to generate an image is proportional to the number of steps used to encode the spatial information. In rapid imaging, an array of coil elements and receivers are used to reduce the number of encoding steps required to generate an image. This is done using knowledge of the spatial sensitivity of the array and receiver channels. Recently, these arrays have begun to include a large number of coil elements. Ideally, each coil element would have its own receiver channel to acquire the image data. In practice, this is not always possible due to economic or other constraints. In this dissertation, methods are explored to combine a large array to a limited number of receivers so as to optimize the performance for parallel imaging; this dissertation focuses on SENSE in particular. Simple combinations that represent larger coils that might be constructed are discussed. More complex solutions form current sheets. One solution uses Roemer's method to optimize image SNR at a set of points. In this dissertation, Roemer's solution is generalized to give the weighting coefficients that optimize SNR over regions. Also, solutions fitted to ideal profiles that minimize noise amplification are shown. These fitted profiles can allow the SENSE algorithm to function at optimal reduction factors. Finally, a description of how to build the combiner in hardware is discussed.

DEDICATION

To Family and Friends who stuck with me all these years.

ACKNOWLEDGEMENTS

I’d like to acknowledge my parents for all their support during this time.

NOMENCLATURE

A, B, C	Real or Complex Scalar
$\vec{A}, \vec{B}, \vec{C}$	Directional Vector
a, b, c	Column or Row Vector
$\hat{\mathbf{x}}$, $\hat{\mathbf{y}}$, $\hat{\mathbf{z}}$, $\hat{\mathbf{n}}$	Unit Vectors
A, **B**, **C**	Matrices
m, n, o, p	Indices from 1…M, 1…N, 1…O, etc…
A^*, B^*, C^*	Complex Conjugate
$\vec{A}^\dagger, \mathbf{a}^\dagger, \mathbf{A}^\dagger$	Conjugate Transpose
MRI	Magnetic Resonance Imaging
SNR	Signal to Noise Ratio
FOV	Field of View
FID	Free Induction Decay
TR	Repetition Time
RF	Radiofrequency

TABLE OF CONTENTS

CHAPTER Page

LIST OF TABLES

Page

LIST OF FIGURES

Page

Page

Page

Page

CHAPTER I

INTRODUCTION

MRI is a noninvasive imaging technique that basically maps the density of proton, hydrogen nuclei, in the patient. By making use of a large static magnetic field, a net magnetization is created within the human body (1). A combination of radiofrequency (RF) magnetic fields and static field gradients are used to manipulate this magnetization and encode an image using Fourier techniques (2). Because Fourier techniques are usually used, the image is essentially acquired one line at a time. The amount of time between acquisitions, the repetition time, is determined by the desired signal to noise ratio (SNR) and contrast in the image (2). The repetition can range from hundreds of milliseconds to seconds making the total imaging time for a 256 line image on the order of seconds to minutes.

During the image acquisition, the patient must remain still to avoid creating motion artifacts in the image (2 pp. 669-701,3,4). For cardiac and torso imaging, the patient must maintain a breath hold during the scan (2 p. 737) —the image acquisition is cardiac gated to minimize effects of cardiac motion (2 pp. 735-737). These requirements can make a scan difficult for the patient, particularly if the patient is already critically ill. Furthermore, the limited size of the magnet bore causes claustrophobia in many people requiring them to be slightly sedated in order to make them comfortable within the magnet. It would obviously be beneficial to decrease the imaging time so as to increase patient comfort. A second economic benefit is by decreasing the scan time required for a patient, more patients can be imaged in a given day. This increased throughput reduces the overall cost of a scan and increases the availability of MRI. These benefits are achieved using rapid imaging techniques.

This dissertation follows the style of Magnetic Resonance in Medicine.

Problem

By using an array of receiver coils to acquire the MRI signal, the SNR of the image acquisition can be increased with proper combination of the independent channels(5,6). Multiple coil systems, with roughly 4 channels, were available in the mid 1990s and were primarily used to improve image quality. In the late 1980s, it was realized that the array of coils could be used to spatially encode the signal received from the body and thereby allow acquisition lines to be skipped thereby reducing total imaging time(5,7). The factor by which the imaging time could be reduced is, theoretically, equal to the number of independent receiver channels used to acquire the image. These techniques are commonly referred to as parallel imaging methods and were not implemented until the late 1990s when the technology became available.

In practice, the reduction factors are much less due to noise amplification caused by the reconstruction algorithm. The algorithm basically uses knowledge of the receiver channel sensitivities to form a system of equations that can be solved (8) to reconstruct the final image. For low reduction factors, the system of equations is over-determined and well conditioned. At high reduction factors, the system of equations may become poorly condition resulting in noise amplification and loss of SNR. The amount of noise amplification in a SENSE reconstruction is quantified by the geometry factor, or g-factor(8).

Much of the development in arrays for use with rapid imaging, in particular SENSE imaging(8), has been to design arrays that minimize the noise amplification while maximizing the reduction factor. Initially, this was done by using array comprised of elements with localized sensitivities(9-12). In this configuration, each coil element is responsible for a region of the image and the signals are combined optimally to reconstruct the complete image. Another possible array design is to have the array composed of elements that have sensitivity over the entire imaging volume(13-15). These elements are referred global elements since they have global sensitivity.

Between the late 1990s and 2005, technology continued to rapidly advance. Currently, receivers with up to 32 channels are commercially available(16) as well as

arrays of up to 76 elements(17). Research groups have gone further with receivers of up to 64 or 96 channels(18,19) and arrays of with as many as 92 elements(19). This has forced the problem considered here to evolve. Instead of trying to design an optimal array for a limited number of receiver channels, the question is shifting to how many channels are really necessary.

Contribution

Initially, this dissertation was to examine two array topologies, global and local, in order to determine which was best suited to SENSE imaging. The problem is made more complex due to the sensitivity of the SENSE algorithm on image field of view (FOV) and the phase encoding direction(20). Searching for an analytical solution, the problem was attacked in the spectral domain since the image FOV and encoding direction could be directly compared with the coils k-space sensitivity. Rotations in encoding direction would simply be different projections of the coils' point spread function.

The general analytical solution ultimately proved intractable. Determining the array performance for SENSE imaging in the spectral domain seemed tantalizingly close, but a way to generalize the solution to arbitrary arrays, particularly when more than one image acquisition line is used, could not be found. When a single line is acquired, as in SEA imaging(21), the relationship between the array elements spectral sensitivities and the image parameters, FOV and encoding direction, becomes obvious and the structure of the matrix to reconstruct the full image is clearly evident. However, when multiple encoding lines are acquired, the reconstruction matrix takes on a block structure and the interaction between blocks is at least as important as the internal structure of a given block. It seems as though there should exist a recursive relationship within the matrix that could be exploited to determine the array performance, but this could not be found and finally the analytical approach was abandoned.

The next approach to determining the optimal array topology was computational. In order to examine the properties of arrays composed of global or local elements, a computer model was created to simulate these arrays. This model is discussed in detail

in Chapter III. The model uses the principle of superposition to construct coils based on a large base array of smaller simpler coils, such as a grid of loop elements. For example, two square loops of equal size positioned sided by side will combine to form a loop of twice the area if added in phase, but will form a butterfly coil if added in anti-phase. By limiting the choice of currents on the loops in the base array to in-phase, anti-phase, or no current, the continuity of the current in the final configuration will be conserved and the coil can be constructed with a single layer of wire. The number of possible configurations for each channel can be quite large; for a base array of 64 elements, there exists 262144 possible combinations for each channel. In order to allow this model to search for optimal configurations, a genetic algorithm, discussed in Chapter IV, was implemented to explore the large solution space.

While working on the analytical approach to this problem, both the number of available receiver channels and the size of arrays began to increase. It is now possible to buy 32 channel receivers that are modular(16). Several groups are implementing large arrays. The largest commercially available is the TIM array by Siemens(17). Other groups have developed linear arrays with 64 channels (22), (21) and head arrays with 92 channels(19). The 64 channel group constructed their own 64 channel receiver (18) and the 92 channel purchased three Siemen's receivers and stacked them together in parallel(19). Questions in the industry have since become "How many receivers are really necessary?" (23). Generally, the old adage "more is always better" holds true, but at what point are the gains in performance not worth the expense in constructing the hardware?

With the availability of large arrays with similar number of elements to the base arrays used for the genetic algorithm optimization it became possible to actually simulate the algorithm on a real array and let it find an optimal configuration for an imaging protocol using a limited number of channels. However, since the base array already physically exists, it is no longer necessary to limit the choice of weighting coefficients to enforce continuity of current. It is now possible to use complex weighting coefficients for the combination of the base array elements and effectively

create a current sheet for each receiver channel. The problem now is to choose the weighting coefficients to optimize the performance of the array for SNR and for SENSE.

A set of functions was created in Matlab® to explore the combination of an array with a large number of elements for a receiver with a small number of channels. The functions compute SNR maps and predict the noise amplification, g-factor, caused in rapid imaging for planar arrays using any choice of complex weighting coefficients. Arbitrary field of views and phase encoding directions are also allowed. These tools are discussed in Chapter IV.

The solution space for the weighting coefficients necessary to combine an array can be quite large. A 64 element array combined into eight channels, has 1024 degrees of freedom—512 coefficients with real and imaginary components. This space is obviously too large for an efficient search. In order to constrain the problem, the array elements were combined so that each receiver channel would yield the optimal SNR at a different point. This reduces the number of free variables and simplifies the optimization. The 64 element array combined into eight channels optimized at eight points now has only 24 parameters or eight coordinates. The weighting coefficients for producing the optimal SNR at a point for an array of coils have been known for a long time and have been credited to Roemer(6), but were first shown by(5). This solution is discussed in Chapter III.

A problem with using points is that there can exist nulls in the sensitivity in the final image since the SNR is may be highly focused on the chosen points. The focusing of sensitivity becomes more pronounced the closer the point approaches the surface of the array. In order to counteract this effect, Roemer's solution was extended so that, instead of giving weighting coefficients for the optimal SNR at a point, it can now generate coefficients that yield optimal SNR over a region. This allows the sensitivity to be diffused for each channel and each channel can be assigned a different region of the image. When combined to form the full image, the null in sensitivity are significantly reduced. This derivation is presented in Chapter III and its application is shown in Chapter V.

Using the Matlab® tools and the methods for optimizing SNR at points and over regions, it became possible to examine questions about how to best use arrays for MRI. In Chapter VI, the question of how many receiver channels are really necessary to attain a nominal SNR is explored. Also, the effects of using point and region optimized weights for SENSE imaging is examined.

Finally, an examination of what would be required to construct a physical hardware combiner was done. The advantage of building the hardware combiner is that it would allow systems currently in hospitals, typically eight channels at most, to use the large arrays that are being developed. Fixed hardware combiners as well as a variable combiner were considered. The variable combiner would yield the flexibility needed to optimize the base array for different imaging protocols and slices. The quality of the reconstruction would be determined by the quality of components. The effects of resolution and range of the digital components on the quality of the image reconstruction were examined, followed by a statistical analysis of the error in the control. Finally, for the device to be effective, the channels need to be well isolated. The effects of coupling between branches in the combiner were examined. The basics of the design and its requirements are discussed in Chapter VI.

Summary

The contributions made by this dissertation are:

- A genetic algorithm for searching optimal realizable array configurations.
- A set of Matlab® tool for examining optimal current sheets for SENSE and SNR.
- An extension to Roemer's solution to compute weighting coefficients that optimize SNR over a given region instead of at a point.
- A relation for determining the number of receiver channels required given imaging depth and image FOV.
- An discussion of how SENSE imaging is affected by array combination.
- A design and analysis for a hardware combiner.
- The best array for MRI is one that is large and can be flexibly combined to form

optimal current sheets for any application.

Layout

This dissertation is broken down into seven chapters. These chapters are:

- Introduction
- Background
- Methods
- Implementation
- Results
- Discussion
- Conclusions

Chapter II, Background, discusses the basic theory behind the electromagnetic models and magnetic resonance imaging. The chapter is divided into three sections. The first section, *Electromagnetics*, discusses how the coil sensitivities of the array elements, as well as array properties such as self and mutual resistance, mutual reactance, and noise correlation, are computed. The second section, *Imaging*, offers a brief explanation of the origin of the MRI signal and the noise in an imaging experiment and defines the signal to noise ratio (SNR). This section covers the basics of the Fourier encoding used to acquire an image and how an image is reconstructed from the encoded data. This section also talks about the time required to acquire an image and to reconstruct the image. The third section, *Rapid Imaging*, discusses rapid imaging and, in particular, SENSE imaging.

Chapter III, Methods, deals with how an array can be combined to provide different sensitivities for a set of receiver channels and, in effect, construct a virtual array. Mathematically, this is represented by simple matrix operations. The matrix containing the sensitivities of the physical coils are operated on by matrices containing the weighting coefficients. The weighting coefficients for a given channel can be chosen to combine the elements simply, to produce the optimal SNR for the array at a point, to

optimize SNR over a region.

The software models created for examining how arrays are to be combined are described in Chapter IV, Implementation. This chapter is split into two sections; the first details the software written in C++, the second section discusses the Matlab® routines. The *C++* section contains two algorithms. The first is a program to compute the mutual resistance for an array using quasi-static assumptions. The second algorithm implements the array combination where continuity of current is enforced and provides a genetic algorithm to search the solution space to find the optimal coils for maximizing SNR or minimizing g-factor. The *Matlab®* section describes the tools used to examine the arbitrary combination of array elements to a limited number of channels and model a hardware implementation of the combiner. Among these tools are functions to compute SNR maps, to compute g-factor maps, to reconstruct acquired image data, and to calculate weighting coefficients.

Chapter V, Results, is subdivided into four sections: *Validation, Simple Array Combination, Point Combined Arrays,* and *Region Combined Arrays.* The *Validation* section compares results computed using the Matlab® and C++ algorithms created for this dissertation with measured results and previously computed results. The electromagnetics models are compared to measured coil sensitivities. The array combination tools are used to recreate previously published results (24). The section *Simple Array Combination* shows results for combinations where continuity of current is enforced. The resulting combinations for each channel can easily be reconstructed as a single coil. This section examines the differences between global and local array topologies. The next section, *Point Combined Arrays*, contains results where the large array has been combined using Roemer's method to generate the optimal SNR at chosen points. Plots showing conformity and maximum deviation from optimal SNR as imaging depth is increased are computed for various numbers of receiver channels. This section also contains reconstructed images using a point combined arrays using both normal and SENSE methods.

Chapter VI, Discussion, deals predominantly with the hardware implementation

of the signal combiner, but also includes sections on computation time and how to choose points for optimization. Block diagrams of both fixed and variable combiners are included. In either case, the channels within the combiner need to be well isolated, therefore the effects of isolation are examined. For the variable combiner, the resolution required for the digital attenuators and phase shifters is examined. Also, a statistical analysis of the performance of the combiner due to the tolerances of the components is done. Finally, the section includes a discussion of how fast the weighting coefficients can be switched and possible applications.

The final chapter, Chapter VII, Conclusions, summarizes the work in this dissertation, states the conclusions found from this work, and proposes some ideas for future work.

CHAPTER II

BACKGROUND

This chapter provides a brief overview of the electromagnetic and MR imaging theory used in this dissertation. It is split into three sections, electromagnetics(25-29), imaging(1,2,30-33), and SENSE imaging(8,10,34,35). Each of these sections detail what was pertinent for this dissertation and provide references for further information.

Electromagnetics

This section describes how the coil sensitivities and mutual resistance matrices for the arrays of coil elements were computed. Since MRI takes place very near the coil element, in a region typically much smaller than a wavelength, quasi-static approximations were used. These approximations are valid for low frequencies and small arrays. The mutual reactance between array elements is typically removed using various decoupling methods (6,7,36-38) and can safely be ignored. This leaves only the mutual resistance matrix as the dominant source of noise in the imaging acquisition.

Coil Sensitivities

The magnetic vector potential, $\vec{A}$, due to a conducting wire in free space is defined as

$$\vec{A}(\vec{x}) = \frac{\mu_0}{4\pi} \oint_{\ell'} \frac{I(\vec{x}')e^{jk|\vec{x}-\vec{x}'|}}{|\vec{x}-\vec{x}'|} d\vec{x}' \qquad [2.1]$$

where $\vec{x}$ is the observation point, $\vec{x}'$ is a point on the conductor, $k = 2\pi/\lambda$, ℓ' is the path of the conductor, and $I(\vec{x}')$ is the current on the conductor at that point. This is the full wave definition of magnetic vector potential. In MRI, the dimensions of the sample being imaged are typically much smaller than the wavelength, λ. Under this condition, $e^{jk|\vec{x}-\vec{x}'|} \to 1$ and the fields can be considered to behave quasi-statically. The magnetic vector potential then becomes

$$\vec{A}(\vec{x}) = \frac{\mu_0}{4\pi} \oint_{\ell'} \frac{I(\vec{x}')}{|\vec{x} - \vec{x}'|} d\vec{x}' \qquad [2.2]$$

where I is now a constant, uniform current.

The electric and magnetic fields produced by the wire can be found from the magnetic vector using

$$\vec{B} = \nabla \times \vec{A} \qquad [2.3]$$

and

$$\vec{E} = -j\omega\vec{A} - j\frac{1}{\omega\mu\varepsilon}\vec{\nabla}\left(\vec{\nabla}\cdot\vec{A}\right). \qquad [2.4]$$

However, under the quasi-static assumption the divergence term, $\vec{\nabla}\left(\vec{\nabla}\cdot\vec{A}\right)$, becomes negligibly small and can be ignored leaving

$$\vec{E} = -j\omega\vec{A}. \qquad [2.5]$$

Since, the magnetic fields are assumed to behave statically, it is simpler to compute them directly using Biot-Savart (28 pp. 175-78),

$$\vec{B}(\vec{x}) = \frac{\mu_0}{4\pi} \oint_{\ell'} \frac{Id\vec{x}' \times (\vec{x} - \vec{x}')}{|\vec{x} - \vec{x}'|^3}, \qquad [2.6]$$

instead of integrating to find the magnetic vector potential and then computing the curl of $\vec{A}$.

Resistance

There are several mechanisms for losses in MRI. These mechanisms are losses due to the sample, losses due to resistance of the wire, and losses due to radiation. In MRI, the sample is usually contained within a conductive bore which shields the experiment from external radiation and reduces the radiation losses to effectively zero.

Self Resistance

The power dissipated into a homogeneous conductive sample is

$$P_{sample} = \sigma \iiint_V \vec{E}(\vec{x}) \cdot \vec{E}(\vec{x})^* d\vec{x} \qquad [2.7]$$

where σ is the sample conductivity. Substituting Eq. [2.5] into this, it is rewriting in terms of the magnetic vector potential,

$$P_{sample} = \sigma\omega^2 \iiint_V \vec{A}(\vec{x}) \cdot \vec{A}(\vec{x})^* d\vec{x} \qquad [2.8]$$

Recalling that power is also defined as

$$P = \frac{1}{2} I^2 R, \qquad [2.9]$$

then

$$R_{sample} = 2\sigma\omega^2 \iiint_V \vec{A}(\vec{x}) \cdot \vec{A}(\vec{x})^* d\vec{x} \qquad [2.10]$$

assuming the magnetic vector potential, $\vec{A}$, is calculated using a unit current.

The resistance of a conductive wire is defined as

$$R_{wire} = \frac{l}{\sigma_c A} \qquad [2.11]$$

where l is the length of the wire, A is the cross-sectional area of the wire, and σ_c is the conductivity of copper (39 p. 38). This is valid at low frequencies where the current flows through the entire wire cross section. At higher frequencies, the current tends to flow only along the surface of the conductor. The depth to which the current penetrates the conductor is the skin depth (27 p. 54),

$$\delta_s = \sqrt{\frac{2}{\sigma_c \mu \omega}}. \qquad [2.12]$$

For a round wire, the resistance becomes

$$R_{wire} = \frac{l}{\sigma_c \delta_s 2\pi r_w} \qquad [2.13]$$

where r_w is the radius of the wire. For a wire strip, the resistance is

$$R_{wire} = \frac{l}{\sigma_c 2w\delta_s} \qquad [2.14]$$

where w is the width of the conducting strip.

Mutual Resistance

When more than two coils are present, there will be a resistance between them due to eddy currents produced in a conductive sample. The mutual resistance, **R**, between two coils is calculated as

$$R_{ij} = \sigma\omega^2 \iiint_V \vec{A}_i(\vec{x}) \cdot \vec{A}_j(\vec{x})^* \, d\vec{x} \qquad [2.15]$$

where $\vec{A}_i$ and $\vec{A}_j$ are the magnetic vector potentials of the i^{th} and j^{th} coils respectively, and V is the volume of the sample. Calculating for an entire array generates a mutual impedance matrix,

$$\mathbf{R}_{sample} = \begin{bmatrix} R_{11} & R_{12} & \cdots & R_{1N} \\ R_{21} & R_{22} & & R_{2N} \\ \vdots & & \ddots & \vdots \\ R_{N1} & R_{N2} & \cdots & R_{NN} \end{bmatrix}. \qquad [2.16]$$

Since the sample is usually a reciprocal media (25 pp. 116-20), this matrix is hermitian, $\mathbf{R} = \mathbf{R}^T$, and is positive-definite, $eig(\mathbf{R}) > 0$ (40 p. 402). This matrix only includes sample losses. The complete resistance matrix for the array is given by adding the copper losses of the array elements to the main diagonal,

$$\mathbf{R} = \begin{bmatrix} R_{11} & R_{12} & \cdots & R_{1N} \\ R_{21} & R_{22} & & R_{2N} \\ \vdots & & \ddots & \vdots \\ R_{N1} & R_{N2} & \cdots & R_{NN} \end{bmatrix} + \begin{bmatrix} R_{wire} & 0 & \cdots & 0 \\ 0 & R_{wire} & & 0 \\ \vdots & & \ddots & \vdots \\ 0 & 0 & \cdots & R_{wire} \end{bmatrix}. \qquad [2.17]$$

Mutual Reactance

The mutual reactance between coil elements in an array has two components, mutual inductance and mutual capacitance. Mutual inductance measures the degree to which the magnetic fields of two coils couple. Mutual capacitance quantifies how much the electric fields of the array elements couple. In MRI, the elements in an array are designed to be sensitive to magnetic fields and to be insensitive to electric fields. Therefore, inductive coupling is the primary source of coupling between array elements.

For two coils in free space and having wavelengths large, a factor of ten, relative to the dimensions of the array, the mutual reactance, Z_{ij}, is

$$Z_{ij} = j\omega M_{ij} = j\omega \oint_{\ell_j} \vec{A}_i\left(\vec{x}_j\right) d\vec{x}_j \,. \qquad [2.18]$$

Inserting the definition of the magnetic vector potential, Eq. [2.2], the coupling coefficient between the two coil elements becomes

$$M_{ij} = \frac{\mu}{4\pi} \oint_{\ell_i} \oint_{\ell_j} \frac{d\vec{x}_i d\vec{x}_j}{\left|\vec{x}_i - \vec{x}_j\right|}, \qquad [2.19]$$

where ℓ_i and ℓ_j are the conductor paths of the ith and jth coil elements.

Capacitive coupling between elements can be approximated, for low frequencies, as a plate capacitor with capacitance

$$C = \varepsilon \frac{A}{d}; \qquad [2.20]$$

where A is the area of the conductor overlap of the array elements and d is the width of the gap between the conductors in the overlap. The capacitance is greatest where the coil elements are closest together. By ensuring sufficient spacing between the conductors, the capacitance can be maintained at negligible levels and is therefore ignored in this dissertation.

Matching Networks

In order to efficiently transfer power from the coil to the receiver, the

impedances of the devices need to be matched to a common value. Typically, the preamplier is designed to have a minimum noise figure when the coil is matched to 50 ohms. The matching is achieved by transforming the self resistance of the coil to 50 ohms using an LC network. Assuming the components in the network are lossless, the circuit acts as a transformer and simply scales the voltage at the terminals of the coil. The scaling constant is proportional to the square root of the desired resistance divided by the self resistance of the coil, R_c. The voltage at the output of the matching network is

$$v_m = \sqrt{\frac{R_0}{R_c}} v_c . \qquad [2.21]$$

Noise Correlation

The mutual resistance matrix, **R**, is often represented as a noise correlation matrix, Ψ, where the main diagonal of the matrix has been normalized to one. The conversion from resistance matrix to noise correlation matrix is done by

$$\Psi = \begin{bmatrix} \sqrt{R_{11}} & 0 & \cdots & 0 \\ 0 & \sqrt{R_{22}} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \sqrt{R_{N_c N_c}} \end{bmatrix}^{-1} \mathbf{R} \begin{bmatrix} \sqrt{R_{11}} & 0 & \cdots & 0 \\ 0 & \sqrt{R_{22}} & 0 & \vdots \\ \vdots & 0 & \ddots & 0 \\ 0 & \cdots & 0 & \sqrt{R_{N_c N_c}} \end{bmatrix}^{-1} . \qquad [2.22]$$

If the array has been matched to the receiver impedance, then the coil self resistances are identical and the noise correlation matrix is just the mutual resistance matrix scaled by the inverse of the receiver impedance,

$$\Psi = \frac{1}{R_0} \mathbf{R} . \qquad [2.23]$$

Imaging

Magnetic Resonance Imaging (MRI) is an outgrowth of Nuclear Magnetic Resonance (NMR) spectroscopy. In NMR, the frequency data received from a sample is

used to determine the chemical composition and structure of that sample (32). However, in MRI, it was realized that by applying a set of field gradients across the sample, the spatial information of the sample could be encoded in the frequency information of the acquired signal (41). In this section, the origin of the MRI signal is briefly described as well as its interaction with externally applied static and radiofrequency magnetic fields; more detailed information on the origin of MRI signal can be found elsewhere (1,32,42). Using these externally applied fields, the spatial information of the sample is encoded using Fourier techniques (2). Finally, the image is reconstructed with a simple inverse Fourier transform of the acquired data. The amount of time required to acquire an image is also discussed in this section.

Signal Origin

In MRI, a strong static magnetic field, $\vec{B}_0$, is applied to the sample being imaged. By definition, the direction of the static magnetic field defines the $\hat{\mathbf{z}}$ axis for magnetic resonance systems. The magnetic field forces nuclei of hydrogen within the sample to either line up along or opposite the applied field creating a magnetization within the sample. The magnetization is related to the static magnetic field by

$$\vec{M}_0 = \frac{\rho\gamma^2\hbar^2}{4kT}\vec{B}_0 \qquad [2.24]$$

where ρ is the density of hydrogen per unit volume, γ is the gyromagnetic ratio of hydrogen ($2\pi \cdot 42.56E6\, rad/(s \cdot T)$), k is Boltzmann's constant, $\hbar$ is Planck's constant, T is the temperature of the sample in degrees Kelvin, and B_0 is the of the applied static magnetic field in Tesla. For water at room temperature (300K), this magnitude of the magnetization reduces to

$$M_0 = 3.18 \cdot 10^{-3} \left|\vec{B}_0\right| \Delta V \,. \qquad [2.25]$$

This is the magnitude of the magnetization at thermodynamic equilibrium when it is aligned with the static magnetic field. In order to generate an MR image, the magnetization needs to be manipulated.

Bloch Equation

The equation describing the motion of the magnetization over time is called the Bloch equation. It is

$$\frac{d\vec{M}}{dt} = \gamma\left(\vec{M} \times \vec{B}_0\right) + \frac{1}{T_1}\left(M_0 - M_z\right)\hat{\mathbf{z}} - \frac{1}{T_2}\vec{M}_\perp \qquad [2.26]$$

where $\vec{M}$ is the magnetization vector, $\vec{B}_0$ is the static magnetic field, and T_1 and T_2 are relaxation constants. The solution to this differential equation is

$$M_x(t) = e^{-t/T_2}\left[M_x(0)\cos(\omega_0 t) + M_y(0)\sin(\omega_0 t)\right] \qquad [2.27]$$

$$M_y(t) = e^{-t/T_2}\left[M_y(0)\cos(\omega_0 t) - M_x(0)\sin(\omega_0 t)\right] \qquad [2.28]$$

$$M_z(t) = M_z(0)e^{-t/T_1} + M_0\left[1 - e^{-t/T_1}\right], \qquad [2.29]$$

where ω_0 is the frequency at which the magnetization precesses about the applied magnetic field and is defined as

$$\omega_0 = \gamma B_0 . \qquad [2.30]$$

The signal detected in the MR experiment is due to the transverse magnetization,

$$M_\perp(t) = M_x(t) + jM_y(t) = M_\perp(0)e^{j\omega_0 t}e^{-t/T_2}, \qquad [2.31]$$

after it has been excited and is returning to equilibrium.

Signal Voltage

The voltage at the terminals of an RF coil can be shown to be

$$v_{sig}(t) = -\frac{d}{dt}\iiint_V \vec{B}(\vec{x}) \cdot \vec{M}(\vec{x},t)^* \, d\vec{x} . \qquad [2.32]$$

Recall from the Bloch equations [2.27]-[2.29], that the transverse component of the magnetization varies sinusoidally at a frequency of ω_0, while the longitudinal component of the magnetization changes at a much slower rate of $1/T_1$. Since the magnitude of the voltage at the terminals is proportional to the derivative of the

magnetization with respect to time, the contribution of the longitudinal component to the signal voltage is negligible and can be ignored. The effective sensitivity of the of the coil can then be expressed as

$$B_{eff}\left(\vec{x}\right) = \vec{B}\left(\vec{x}\right)\cdot\hat{\mathbf{p}} \qquad [2.33]$$

where

$$\hat{\mathbf{p}} = \frac{1}{\sqrt{2}}\left(\hat{\mathbf{x}} + j\hat{\mathbf{y}}\right) \qquad [2.34]$$

is the unit vector denoting the polarization sensitivity of the RF coil. By substituting Eqs. [2.31] and [2.33] into Eq. [2.32], a scalar equation for the signal voltage is found,

$$v_{sig}\left(t\right) = -\frac{d}{dt}\iiint_V B_{eff}\left(\vec{x}\right) M_{\perp}\left(\vec{x},0\right)^{*} e^{j\omega_0 t} e^{-\frac{t}{T_1}} e^{-\frac{t}{T_2}} d\vec{x}\,. \qquad [2.35]$$

Since the RF frequency, ω_0 is much greater than the relaxation constants, T_1 or T_2, applying the derivate simplifies the signal voltage to

$$v_{sig}\left(t\right) = -j\omega_0\iiint_V B_{eff}\left(\vec{x}\right) M_{\perp}\left(\vec{x},0\right)^{*} e^{j\omega_0 t} e^{-\frac{t}{T_1}} e^{-\frac{t}{T_2}} d\vec{x}\,. \qquad [2.36]$$

Ignoring relaxation effects completely gives

$$v_{sig}\left(t\right) = -j\omega_0\iiint_V B_{eff}\left(\vec{x}\right) M_{\perp}\left(\vec{x},0\right)^{*} e^{j\omega_0 t} d\vec{x}\,. \qquad [2.37]$$

This will be the preferred form for the signal voltage used in this dissertation. For a small volume, assuming the sample and coil sensitivity are homogeneous, the time average signal is equal to

$$v_{sig} = -j\omega_0 B_{eff} M_0 \sin\left(\alpha\right)\Delta V \qquad [2.38]$$

where α is the tip angle to which the system was excited. This becomes,

$$v_{sig} = -j\omega_0\left(\mathbf{B}\cdot\hat{\mathbf{p}}\right)\frac{\rho\gamma^2\hbar^2}{4kT} B_0 e^{-j\phi} \sin\left(\alpha\right)\Delta V \qquad [2.39]$$

when B_{eff} and M_0 are expanded and the initial phase of the magnetization, ϕ, is

included. Since the coil sensitivity and magnetization vary slowly over a voxel, three dimensional pixel, this expression is a good approximation for the signal at the terminals of a coil for a voxel with volume ΔV.

Definition of SNR

The Signal to Noise Ratio (SNR) is a commonly used quantity for measuring the quality of a signal. In communications, this is often defined as the ratio of signal power to noise power. However, in MRI, the SNR is defined as the ratio of signal voltage to noise voltage,

$$SNR = \frac{v_{sig}}{v_{noise}}. \qquad [2.40]$$

Noise in MRI comes from several sources, the thermal noise of the sample, the thermal noise of the coil, and the noise from background radiation. It can be shown that each of these noise sources is associated with a resistance (2 p. 334). The thermal noise voltage generated by a given resistance is

$$v = \sqrt{4kTR\Delta f} \qquad [2.41]$$

where k is Boltzmann's Constant, R is the resistance, T is the temperature of the resistor in Kelvin, and Δf is the bandwidth of the receiver. Therefore, the noise voltages due to the sample, coil, and background radiation are

$$v_{sample} = \sqrt{4kT_{sample}R_{sample}\Delta f}\,, \qquad [2.42]$$

$$v_{coil} = \sqrt{4kT_{coil}R_{coil}\Delta f}\,, \qquad [2.43]$$

and

$$v_{rad} = \sqrt{4kT_{rad}R_{rad}\Delta f} \qquad [2.44]$$

respectively. Typically, all the noise sources are assumed to be at the same temperature, usually 300K, and, consistent with the quasi-static assumptions, R_{rad} is usually much less than R_{coil} or R_{sample} and is ignored, leaving

$$v_{noise} = \sqrt{4kT\left(R_{sample} + R_{coil}\right)\Delta f}\ . \qquad [2.45]$$

Combining Eqs. [2.45] and [2.39] into [2.40] gives the signal to noise ratio of a image voxel,

$$SNR_{voxel} = \frac{\omega_0\left(\vec{B}\cdot\hat{\mathbf{p}}\right)\frac{\rho\gamma^2\hbar^2}{4kT}B_0\sin(\alpha)\Delta V}{\sqrt{4kT\left(R_{sample}+R_{coil}\right)\Delta f}}, \qquad [2.46]$$

at the terminals of the coil. Ideally, the signal would be acquired directly at this point, but it must be brought out of the bore, pre-amplified, combined, and demodulated before it can be digitized and acquired. All these steps degrade the SNR.

Definition of Noise Figure

The amount to which a component degrades the SNR is quantified by its noise figure, F. The noise figure is a measure of how much noise is added to the signal by that component is a logarithmic expression of noise factor. The noise factor, *f*, of a component is defined as

$$f = \frac{SNR_{in}}{SNR_{out}} \qquad [2.47]$$

where SNR_{in} and SNR_{out} is the SNR at the input and output ports of the device. The noise figure, *F,* of the device is simply the noise factor converted to decibels,

$$F = 10\log(f). \qquad [2.48]$$

A receiver typically contains several components in series to amplify and filter the signal prior to digitization. The noise figure an entire receiver is calculated according to

$$f = f_1 + \frac{f_2 - 1}{g_1} + \frac{f_3 - 1}{g_1 g_2} + \ldots + \frac{f_n - 1}{g_1 g_2 \cdots g_{n-1}} \qquad [2.49]$$

where g is the gain of the component. From this equation, it is easy to show that if the gain of the first component in the receiver is large, then $f \approx f_1$ and the noise factor of that component will dominate the noise figure for the entire receiver. Assuming the

matching networks for a receiver coil are lossless, then the noise figure of the pre-amplifier will be the major factor in defining the receiver noise figure. A typical pre-amplifier used in MRI will have a gain of approximately 30dB with a noise figure of 0.5 dB.

Imaging Basics

Typically, the static magnetic field, $\vec{B}_0$, is homogeneous over the imaging volume. With only this field affecting the spin system, the resulting MR signal, Eq. [2.36], is nearly monotonic and results in a signal called a free induction decay (FID). The FID signal and its associated spectrum are shown in Figure 1. This signal contains no spatial information. In order to generate an image, the spin density of the sample needs to be mapped to position.

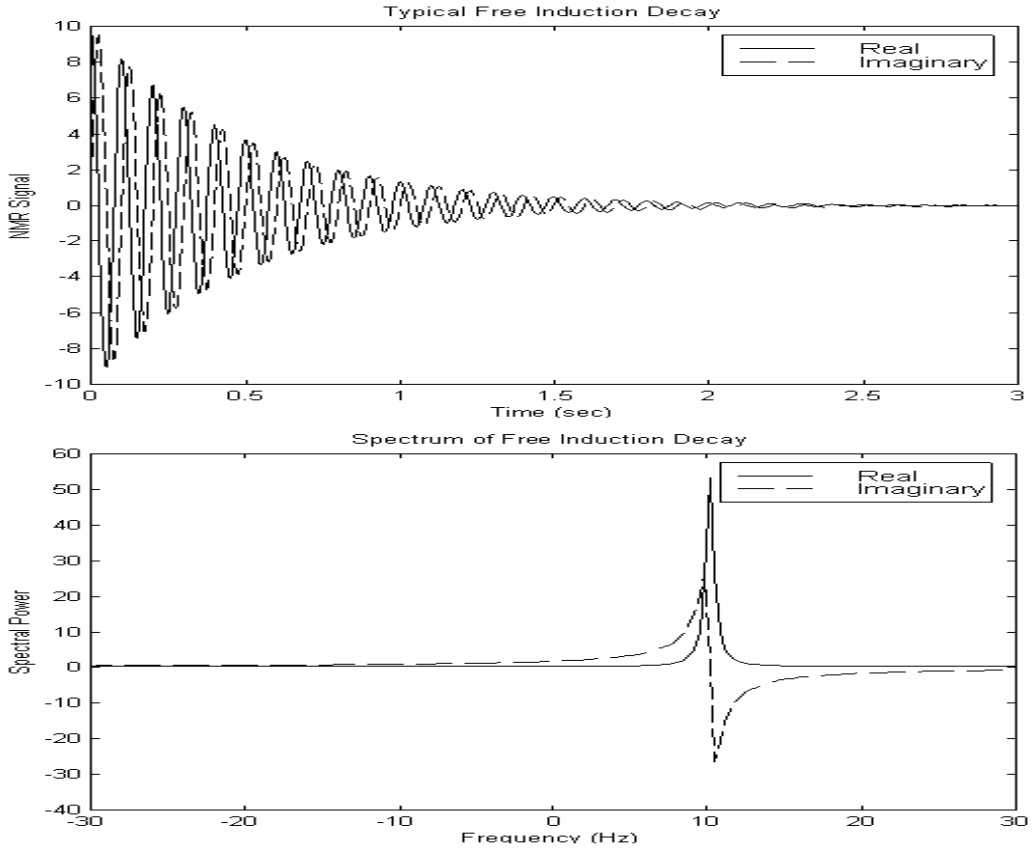


Figure 1. MR signal and associated spectrum.

Application of Gradients

Recall from the Bloch equations, Eqs. [2.27] - [2.29], that the magnetization from the spin system will precess about the applied static magnetic field at the frequency

$$\omega_0 = \gamma \left| B_0 \right| = \gamma B_0 . \qquad [2.50]$$

By superimposing a set of linear magnetic field gradients onto the main static field, the rate at which the magnetization rotates with position can be controlled. The frequency for the MR signal now becomes spatially dependent with

$$\omega(x, y, z) = \gamma \left(B_0 + G_x x + G_y y + G_z z \right) \qquad [2.51]$$

where G_x, G_y, and G_z are the slopes of the field gradients in the $\hat{\mathbf{x}}$, $\hat{\mathbf{y}}$, and $\hat{\mathbf{z}}$ directions. It is implied from this equation that polarization of the magnetic field gradients is in the same direction as the static field; along the z-axis. This equation, Eq. [2.51], may be written more succinctly as the vector product

$$\omega(\vec{x}) = \gamma \left(B_0 + \vec{G} \cdot \vec{x} \right). \qquad [2.52]$$

Frequency Encoding

Including gradients in the equation for the MR signal, Eq. [2.36], gives

$$v_{sig}(t) = -j\omega_0 \iiint_V B_{eff}(\mathbf{x}) M_{\perp}(\mathbf{x}, t)^* \, e^{j\gamma B_0} e^{j\gamma(\mathbf{G}\cdot\mathbf{x})t} e^{-\frac{t}{T_2}} d\mathbf{x} . \qquad [2.53]$$

This formula gives the equation for the MR signal during an acquisition window starting at $t = t_{acq}$ where the sample magnetization has been excited and prepared prior to this time to create spin and gradient echoes (2).

Typically, prior to the signal being acquired, it is demodulated down to baseband. Mathematically this is equal to multiplying the signal voltage by $e^{-j\omega_o t}$ in order to remove the high frequency component yielding

$$v_{sig}(t) = -j\omega_0 \iiint_V B_{eff}(\mathbf{x}) M_{\perp}(\mathbf{x}, t)^* \, e^{j\gamma(\mathbf{G}\cdot\mathbf{x})t} e^{-\frac{t}{T_2}} d\mathbf{x} \qquad [2.54]$$

This form of the MR signal is commonly referred to as being in the rotating frame in the MR literature (2 p. 36). Also, this equation is continuous in time while the acquired signal is usually sampled at N distinct points at a sampling rate of Δt. Including the effects of digitization, the acquired signal is

$$v_{sig}(n) = -j\omega_0 \iiint_V B_{eff}(\vec{x}) M_{\perp}(\vec{x}, n\Delta t)^* e^{j\gamma(\vec{G}\cdot\vec{x})n\Delta t} e^{-\frac{n\Delta t}{T_2}} d\vec{x}. \qquad [2.55]$$

With an applied gradient amplitude of zero, $|\vec{G}| = 0$, the resulting signal is a monotonic FID. When a gradient is applied, the signal begins to contain spatial information about the sample magnetization. Taking the spectrum of the signal yields a profile of the magnetization as shown in Figure 2. Making a profile of the sample in this manner is call frequency encoding.

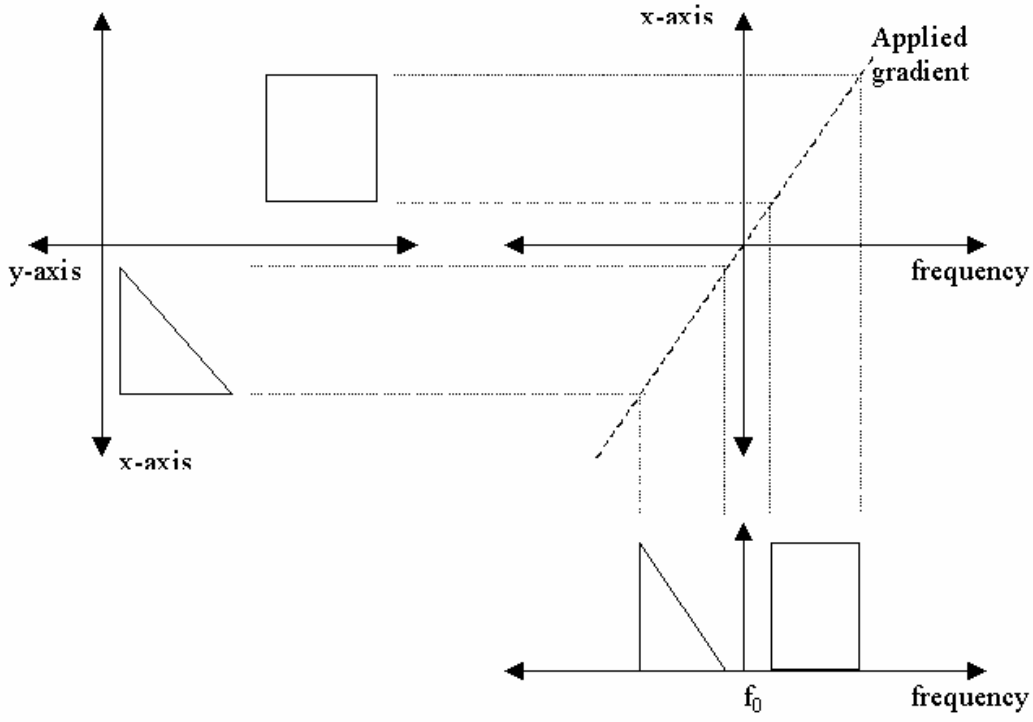


Figure 2. Creating an imaging profile using a gradient.

Typically, the quantity $\gamma\left(\vec{G}\cdot\vec{x}\right)$ is much greater than either T_1 or T_2 meaning that the effect of the gradients makes the relaxation effects negligible so they can be ignored. This simplifies Eq. [2.55] to

$$v_{sig}\left(n\right) = -j\omega_0 \iiint_V B_{eff}\left(\vec{x}\right) M_{\perp}\left(\vec{x}, t_{acq}\right)^* e^{j\gamma\left(\vec{G}\cdot\vec{x}\right)n\Delta t} d\vec{x}\,. \qquad [2.56]$$

At this point, a change of variable is introduced to simplify the formula further. By letting

$$\vec{k} = \gamma\vec{G}\Delta t \qquad [2.57]$$

the equation becomes

$$v_{sig}\left(\vec{k}\right) = -j\omega_0 \iiint_V B_{eff}\left(\vec{x}\right) M_{\perp}\left(\vec{x}, t_{acq}\right)^* e^{j\vec{k}\cdot\vec{x}} d\vec{x}\,. \qquad [2.58]$$

In this form, it becomes obvious that the addition of the linear gradients is encoding the MR signal such that the received signal is the Fourier transform of the magnetization weighted by the coil sensitivity.

The direction in which the signal is frequency encoded is given by the direction of $\vec{k}$ which is also the same as the direction of give by the gradient vector $\vec{G}$. This gradient is commonly called the readout gradient since frequency encoding occurs during the signal digitization. For simplicity, the readout gradient will be assumed to be in the $\hat{\mathbf{x}}$ direction. This reduces Eq. [2.58] to

$$v_{sig}\left(k_x\right) = -j\omega_0 \iiint_V B_{eff}\left(\vec{x}\right) M_{\perp}\left(\vec{x}, t_{acq}\right)^* e^{jk_x x} d\vec{x}\,. \qquad [2.59]$$

During digitization, the k-space is sampled an N discrete points. Since the k-space is related to the image through a Fourier Transform, the image field of view and resolution are dependent on how the k-space is sampled. This relationship is given by the Nyquist relations. Using the Nyquist relations, the FOV is inversely proportional to the rate at which the k-space is sampled,

$$FOV_x = \frac{1}{\Delta k_x} = \frac{1}{\gamma G_x \Delta t}\,. \qquad [2.60]$$

The resolution of the image is simply the FOV of the image divided by the number of points in the image,

$$Res_x = \frac{FOV_x}{N} = \frac{1}{N\Delta k_x} = \frac{1}{\gamma G_x N \Delta t} = \frac{1}{\gamma G_x T_{acq}}. \qquad [2.61]$$

From Eq. [2.61], the relationships between resolution and several variables are shown. It can easily be seen that the resolution of the image is inversely proportional to the extent of which the k-space has been measured.

Care must be taken in choosing the values for the readout gradient G_x and the sampling rate Δt. The sampling rate determines the frequency bandwidth of the image. If the values for G_x or Δt are chosen inappropriately, the field of view of the image will be smaller than the size of the object. When this occurs, the object will appear folded back on itself in the image. This effect is called aliasing. An example of aliasing is shown in Figure 3.

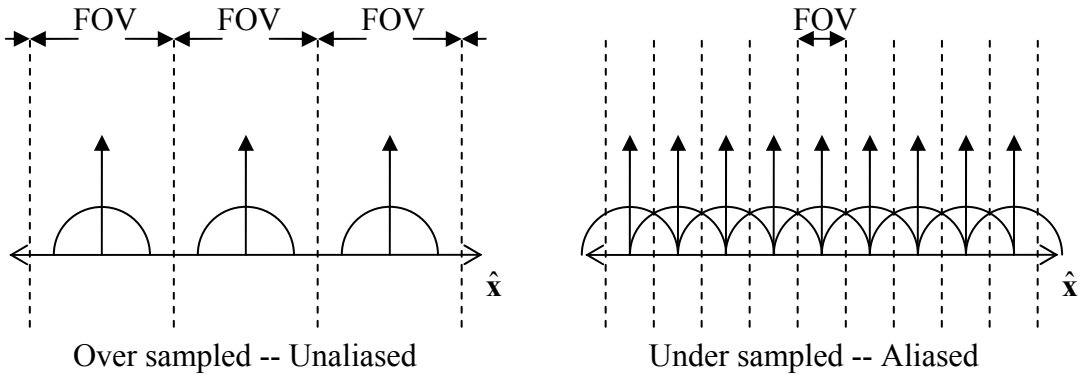


Figure 3. Example of aliasing.

Phase Encoding

Up to this point, the spatial information has only been encoded in the direction of the readout gradient and results in a projection or profile of the object. In order to generate an image, the magnetization needs to be encoded in a second, preferably orthogonal, direction. This direction will be assumed to be parallel to the $\hat{\mathbf{y}}$ axis.

Encoding in this direction is achieved by applying the phase encoding field gradient for a length of time, T_{pe}, prior to the acquisition window. The gradient encodes a phase distribution across the sample that is proportional to position. Including the phase encode gradient into the MR signal gives

$$v_{sig}\left(k_x,k_y\right)=-j\omega_0\iiint_V B_{eff}\left(\vec{x}\right)M_{\perp}\left(\vec{x},t_{acq}\right)^{*}e^{jk_y y}e^{jk_x x}d\vec{x} \qquad [2.62]$$

where

$$k_y=\gamma G_y T_{pe}\,. \qquad [2.63]$$

Like frequency encoding, the field of view and resolution of the image in the phase encode direction is determine by how the k-space is sampled in that direction. The field of view in this direction is given by

$$FOV_y=\frac{1}{\Delta k_y}=\frac{1}{\gamma\Delta G_y T_{pe}}\,. \qquad [2.64]$$

Similarly, the resolution is given by

$$Res_y=\frac{FOV_y}{M}=\frac{1}{M\Delta k_y}=\frac{1}{\gamma M\Delta G_y T_{pe}}\,. \qquad [2.65]$$

Just as in frequency encoding, the choice of phase encode time and the step size of the phase encode gradient should be chosen so that the field of view is larger than the object being imaged, otherwise aliasing will occur. In order to acquire all M phase encode steps, it is necessary to repeat the acquisition M times. The amount of time between acquisitions is call the recycle time or repetition time and is represented by the variable *TR*.

The object has now been encoded in two dimensions resulting in a projection of the object onto the plane that contains both the readout and phase encode unit vectors. For readout in $\hat{\mathbf{x}}$ and phase encode in $\hat{\mathbf{y}}$, this projection is on the axial plane. It is possible to create a full three dimensional image data set by adding another phase encoding step prior to the signal acquisition. The phase encoding step is applied in the z

direction and gives

$$v_{sig}\left(\vec{k}\right) = -j\omega_0 \iiint_V B_{eff}\left(\vec{x}\right) M_{\perp}\left(\vec{x}, t_{acq}\right)^* e^{j\vec{k}\cdot\vec{x}} d\vec{x} \qquad [2.66]$$

where

$$k_z = \gamma G_z T_{pe}. \qquad [2.67]$$

The resolution and field of view is defined just as it was in the y direction;

$$FOV_z = \frac{1}{\Delta k_z} = \frac{1}{\gamma \Delta G_z T_{pe}} \qquad [2.68]$$

and

$$Res_z = \frac{FOV_z}{L} = \frac{1}{L\Delta k_z} = \frac{1}{\gamma L \Delta G_z T_{pe}}. \qquad [2.69]$$

In order to acquire a full three dimensional data set, the acquisition must be repeated *LM* times. This greatly increases the time required to generate an image.

Similar to phase encoding are the dephase gradients used to center the k-space in the acquisition window. Since most of the energy in the MR signal is at low frequencies, it is necessary to center the k-space on the origin. This is done by applying a constant gradient pulse to the magnetization before it is spatially encoded. The equation describing the image data set is then

$$v_{sig}\left(n,m,l\right) = -j\omega_0 \iiint_V B_{eff}\left(\mathbf{x}\right) M_{\perp}\left(\mathbf{x}, t_0\right)^* e^{j\left(k_{z0}+l\Delta k_z\right)z} e^{j\left(k_{y0}+m\Delta k_y\right)y} e^{j\left(k_{x0}+n\Delta k_x\right)x} d\mathbf{x} \qquad [2.70]$$

where

$$k_{x0} = -\frac{\left(N-1\right)\Delta k_x}{2}, \qquad [2.71]$$

$$k_{y0} = -\frac{\left(M-1\right)\Delta k_y}{2}, \qquad [2.72]$$

and

$$k_{z0} = -\frac{(L-1)\Delta k_z}{2} \qquad [2.73]$$

with the indices

$$l = 1...L$$

$$m = 1...M \qquad [2.74]$$

$$n = 1...N .$$

Slice Selection

Slice selection is a method used to limit the projection in the third dimension when only one plane or slice of the object is to be imaged. It has been assumed so far that the excitation of the spin system has been uniform across the sample. Typically, this is achieved by a short excitation RF pulse where the bandwidth of the pulse is much greater than the bandwidth of the sample FID. Since the spectrum of the FID is contained within the spectrum of the RF excitation, the spin system is excited and a transverse magnetization is created everywhere within the object being imaged.

In slice selection, a gradient is applied in the direction perpendicular to the imaging plane so that the spectrum of the sample is now spread out in this direction. The RF pulse is then modulated using a sinc function so that it has a narrow spectrum. Due to the frequency selective nature of a nuclear spin, only those spins whose frequencies are within the narrow spectrum of the RF pulse will be excited. Therefore, the combination of a magnetic field gradient and the narrow band RF pulse will only excite a thin slice of the object. The thickness of the slice is slice is

$$\Delta z = \frac{\Delta f}{\gamma G_z} . \qquad [2.75]$$

The bandwidth of the RF pulse is usually fixed and the thickness of the slice is controlled by the amplitude of the gradient. When using slice selection, the acquired data is

$$v_{sig}(n,m) = -j\omega_0 \int_{z_0-\frac{\Delta z}{2}}^{z_0+\frac{\Delta z}{2}} \iint B_{eff}(\vec{x}) M_{\perp}(\vec{x},t_0)^* e^{j(k_{y0}+m\Delta k_y)y} e^{j(k_{x0}+n\Delta k_x)x} dxdydz . \quad [2.76]$$

Image Reconstruction

The signal that is acquired and digitized is a k-space representation of the image. An example of an acquired image data set is shown in Figure 4. In order to reconstruct the image, the inverse Fourier transform is applied to the acquired data set. Applying the inverse transform to Eq. [2.70],

$$I(\vec{x}) = \iiint \left\{ -j\omega_0 \iiint_V B_{eff}(\vec{x}) M_{\perp}(\vec{x},t_{acq})^* e^{jk_z z} e^{jk_y y} e^{jk_x x} d\vec{x} \right\} e^{-jk_z z} e^{-jk_y y} e^{-jk_x x} d\vec{k} , \quad [2.77]$$

gives

$$I(\vec{x}) = -j\omega_0 B_{eff}(\vec{x}) M_{\perp}(\vec{x})^* . \quad [2.78]$$

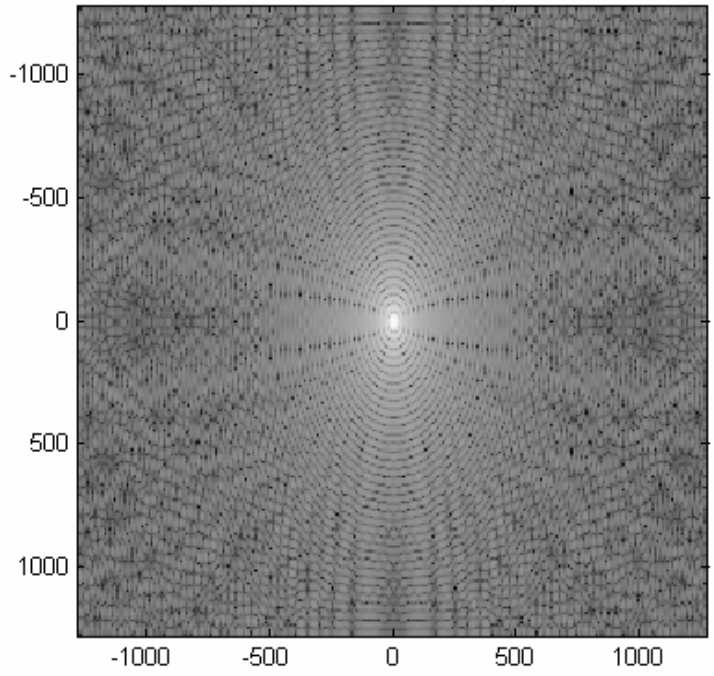


Figure 4. K-Space example in log scale.

However, since the data is actually sampled, the transformation is actually a discrete Fourier transform,

$$I(\vec{x}) = \sum_{l=1}^{L}\sum_{m=1}^{M}\sum_{n=1}^{N}\left\{-j\omega_0 \iiint_V B_{eff}(\vec{x}) M_{\perp}(\vec{x},t_o)^* e^{jk_z z} e^{jk_y y} e^{jk_x x} d\vec{x}\right\} e^{-jn\Delta k_x x} e^{-jm\Delta k_y y} e^{-jl\Delta k_z z}, \quad [2.79]$$

yielding

$$I(\vec{x}) = -j\omega_0 B_{eff}(\vec{x}) M_{\perp}(\vec{x})^* \Delta x \Delta y \Delta z \quad [2.80]$$

for the signal level for a given voxel. This is identical to Eq. [2.38], but now instead of the signal level being produced by the entire sample volume, it is localized and limited to the volume of a single pixel. Since the noise is not spatially dependent, it is uniform throughout the image. The image SNR is then

$$SNR(\vec{x}) = \frac{\omega_0 \left| B_{eff}(\vec{x}) M_{\perp}(\vec{x})^* \right| \Delta x \Delta y \Delta z}{\sqrt{4kT\left(R_{sample} + R_{coil}\right)\Delta f}}. \quad [2.81]$$

Imaging Time

The time it takes to acquire an image is dominated by the repetition time, *TR*. This is the time delay between each different line in the k-space. The repetition time is usually on the order of hundreds of milliseconds or seconds and is chosen to control the contrast in an image between different materials. For a slice selected image, the imaging time is $M \cdot TR$. For a 256x256 imagine with a TR of half a second, the imaging time is 128 seconds or just over two minutes. During this time, the sample must be kept motionless or the image will be distorted. When the sample is a patient, this can be an uncomfortable amount of time.

For a full three dimensional data set, the time required is equal to $L \cdot M \cdot TR$. This duration increase rapidly with the number of planes. For a coarse data set of sixteen planes having 256x256 resolution and a TR of half a second, it takes over half an hour to acquire the image. This is obviously unacceptable for clinical imaging. It is

possible to acquire an entire plane in a single TR using Echo Planar Imaging and similar techniques, but these are beyond the scope of this text.

Reconstruction Time

The information acquired to construct an MR image is encoded in the k-space, Eq. [2.70] for a 3D set or Eq. [2.76] when using slice selection. In order to actually view the image, it must be transformed using the inverse Discrete Fourier Transform. This is done using the Fast Fourier Transform algorithm, the details of which can be found here (2). Using this algorithm, the time it takes to process one line of the k-space is $O\left(n\log\left(n\right)\right)$. The number of operations it takes to evaluate one acquisition window is $N\log\left(N\right)$. In a planar image, there is one acquisition window for each phase encode step. For an M-by-N image, the number of operations required to reconstruct the image is

$$Ops\_2D = M\log\left(M\right) + N\log\left(N\right). \qquad [2.82]$$

Similarly, for a three-dimensional data set, the number of operations required to reconstruct the image is

$$Ops\_3D = L\log\left(L\right) + M\log\left(M\right) + N\log\left(N\right). \qquad [2.83]$$

The actual time required to do the reconstruction is machine dependent. Different computer architectures will perform the operations at different speeds and may require additional operations, such as memory management, that do not directly contribute to reconstructing the images.

Rapid Imaging

Nearly all the time spent acquiring an image is spent phase encoding the spatial information. Long ago, it was realized that an array of RF coils could be used to encode spatial information as well. It wasn't until the late 1990s with the advent of commercially available multiple receiver systems originally intended to use array to increase SNR, did the possibility of rapid imaging begin to be exploited.

Looking back at equation[2.76], it is seen that the image of the magnetization is weighted by the sensitivity of the RF coil. At first look, it seems it would be possible to designate a coil to a subsection of the image and then just piece the complete image together like a puzzle. However, due to the nature of Fourier encoding, this isn't possible, each coil will receive information about the entire image. It is impossible, due to the nature of Maxwell's equations, to simply assign a coil to a specific region.

In 1998, SMASH(43) was presented as a way to use an array as a means to reduce imaging time. In 2000, the SENSE(8) algorithm was introduced. It was followed quickly by PILS(44) and GRAPPA(45). By 2005, the MR industry had settled on SENSE and GRAPPA as the preferred methods for implementing rapid imaging. The focus of this dissertation is on the SENSE algorithm.

Reduction of K-space

In order to reduce the imaging time, all the algorithms reduce the number of phase encode acquisitions. This is by skipping lines in the k-space. For example, a reduction factor of two would acquire lines with index $m = 1,3,5,7,..,M$ and a reduction factor of four would have an index of $m = 1,5,9,13,...,M$. The amount of time saved in the image acquisition is equal to the reduction factor. However, recall from Eq. [2.64] that the field of view in the phase encode direction is determined by the spacing of the sample points in k-space. By skipping lines to reduce imaging time, the field of view of the image has been reduced. The images acquired from the array of coils are therefore aliased.

Unaliasing Images

In the aliased image, a pixel may contain information from more that one pixel in the full image. FIGURE shows how the information becomes overlapped as the reduction factor is increased. At a reduction factor of one, the field of view hasn't been reduced and there is no overlap. At a reduction factor of two, the field of view is reduced and pixels near the edge now contain information from two points. This overlap grows more severe as the reduction factor is increased.

Knowledge of the coil sensitivities over the field of view is used to unalias the individual images and reconstruct the final combined image. Recall from Eq. [2.80] that the signal in a pixel is proportional to the magnetization in the voxel scaled by the coil sensitivity.

$$I\left(\vec{x}\right) \propto B_{eff}\left(\vec{x}\right) M_{\perp}\left(\vec{x}\right)^{*} \qquad [2.84]$$

From now on, in order to simplify the notation, it will be assumed that the pixel signal will be equal to this value. When aliasing occurs, this becomes

$$I\left(\vec{x}\right) = B_{eff}\left(\vec{x}_0\right) M_{\perp}\left(\vec{x}_0\right)^{*} + B_{eff}\left(\vec{x}_1\right) M_{\perp}\left(\vec{x}_1\right)^{*} + \ldots + B_{eff}\left(\vec{x}_{R-1}\right) M_{\perp}\left(\vec{x}_{R-1}\right)^{*} \qquad [2.85]$$

where R is the reduction factor. This is more written more succinctly has a vector product

$$I\left(\vec{x}\right) = \mathbf{B} \cdot \mathbf{M} \qquad [2.86]$$

When an array of coils are used, each coil will acquire an aliased image that will be weighted by that coils sensitivity,

$$I_c\left(\vec{x}\right) = \mathbf{B}_c \cdot \mathbf{M}\,. \qquad [2.87]$$

In SENSE, it is realized that these coil sensitivities can be used to form a system of equations,

$$\mathbf{I} = \mathbf{B} \cdot \mathbf{M}\,. \qquad [2.88]$$

that allows the magnetization in the aliased points to be solved for,

$$\mathbf{M} = \mathbf{U} \cdot \mathbf{I}\,, \qquad [2.89]$$

using an unfolding matrix **U**. The unfolding matrix is chosen so that it maximizes the SNR in a pixel as well as unaliases the pixels. The unfolding matrix is defined to be

$$\mathbf{U} = \left(\mathbf{B}^{\dagger}\mathbf{R}^{-1}\mathbf{B}\right)^{-1}\mathbf{B}^{\dagger}\mathbf{R}^{-1}\,. \qquad [2.90]$$

The derivation of this matrix is discussed later in the methods section on array combination.

From Eq. [2.88], it can be easily seen that the in order to be able to solve for **M**, the reduction factor must always be less than or equal to the number of coils in the array, otherwise the system of equations will be under determined. When the reduction factor is less than the number of coils, the system is said to be over determined and the unfolding matrix will automatically use these extra degrees of freedom to optimize the SNR in the pixels in a least squares sense. The SNR in the combined image is described by

$$SNR_{combined} = \frac{SNR_{full}}{g\sqrt{R}} \qquad [2.91]$$

where g is the g-factor associated with the array. The SNR is inherently reduced by the square root of the reduction factor since less information is acquired to generate the image.

Noise Amplification

The g-factor is a position dependent quantity that describes how the noise was amplified in a given pixel during the process of unaliasing and combining the acquired images. The g-factor is related to the condition of the unfolding matrix and is defined as

$$g_{(x,y)} = \sqrt{\left(\mathbf{B}^{\dagger}\mathbf{R}^{-1}\mathbf{B}\right)^{-1}_{(x,y),(x,y)} \cdot \left(\mathbf{B}^{\dagger}\mathbf{R}^{-1}\mathbf{B}\right)_{(x,y),(x,y)}} \geq 1 . \qquad [2.92]$$

The g-factor will always be greater than or equal to one.

Optimally, the g-factor would be uniformly equal to one over the entire FOV and means that noise is not amplified as a result of the SENSE reconstruction. Ignoring mutual resistance and expanding the matrix **B** using a singular value decomposition (40 p. 157),

$$\mathbf{B} = \mathbf{U}^{\dagger}\mathbf{S}\mathbf{V} , \qquad [2.93]$$

the g-factor can be rewritten as

$$g_{(x,y)} = \sqrt{\left(\mathbf{V}^{\dagger}\mathbf{S}^{\dagger}\mathbf{U}\mathbf{U}^{\dagger}\mathbf{S}\mathbf{V}\right)^{-1}_{(x,y),(x,y)} \cdot \left(\mathbf{V}^{\dagger}\mathbf{S}^{\dagger}\mathbf{U}\mathbf{U}^{\dagger}\mathbf{S}\mathbf{V}\right)_{(x,y),(x,y)}} . \qquad [2.94]$$

By definition of the singular value decomposition, the matrices **V** and **U** are unitary,

$\mathbf{V}^{\dagger}\mathbf{V} = \mathbf{1}$ and $\mathbf{U}^{\dagger}\mathbf{U} = \mathbf{1}$, and the matrix **S** is diagonal with the singular values, $\sigma_{(x,y)}$, along the main diagonal. Using these properties, the g-factor is reduced to

$$g_{(x,y)} = \sqrt{\sum\left(\frac{1}{\sigma^2_{(x,y)}}\mathbf{V}^{\dagger}_{(x,y)}\mathbf{V}_{(x,y)}\right)\cdot\sum\left(\sigma^2_{(x,y)}\mathbf{V}^{\dagger}_{(x,y)}\mathbf{V}_{(x,y)}\right)}. \qquad [2.95]$$

Examining this equation, it is seen that the g-factor can only be made equal to one over the entire image in two ways. First, all the singular values can be identical allowing them to be factored out of the summations,

$$g_{(x,y)} = \sqrt{\frac{\sigma^2_{(x,y)}}{\sigma^2_{(x,y)}}\sum\left(\mathbf{V}^{\dagger}_{(x,y)}\mathbf{V}_{(x,y)}\right)\cdot\sum\left(\mathbf{V}^{\dagger}_{(x,y)}\mathbf{V}_{(x,y)}\right)}, \qquad [2.96]$$

and the summations then reduce to the identity matrix since **V** is, by definition, unitary,

$$g_{(x,y)} = \sqrt{1\cdot\mathbf{1}\cdot\mathbf{1}} = 1 \qquad [2.97]$$

and the g-factor becomes uniformly equal to one.

The second method for optimizing g-factor is to construct the array so the **V** is diagonal. This occurs when $\mathbf{v}_i \cdot \mathbf{v}_j^{\dagger} = \delta_{ij}$ and means that **V** will only select the singular value corresponding to its position in the summation and, since **V** is also unitary, Eq. [2.95] becomes

$$g_{(x,y)} = \sqrt{\left(\mathbf{S}^{\dagger}\mathbf{S}\right)^{-1}_{(x,y),(x,y)}\cdot\left(\mathbf{S}^{\dagger}\mathbf{S}\right)_{(x,y),(x,y)}} = \mathbf{1} \qquad [2.98]$$

The choice of which method to use depends on which is more practical to implement. Finding the array that sets all the eigenvalues for every pixel in an image to be equal can be exceedingly difficult. It may be easier to find an array that satisfies the second method.

CHAPTER III

METHODS

In order for an array to be used to generate an image for MRI, its outputs must be combined. The combination is usually done in software, but can be done in hardware as well.

Array Combination

Array combination can be implemented in either hardware or software. The advantage of building a hardware combiner is that fewer receiver channels are required. If implemented in software, each array element would require its own receiver channel. The effect of the combination would be to reduce the amount of time required to reconstruction the image. Regardless of which method is chosen, the effectiveness of the combined array depends on how it is combined. In this section, the composite coil sensitivities and effective resistance matrix are defined given a set of weighting coefficients. The methods for choosing appropriate weighting coefficients are discussed in the next section.

Composite Coil Sensitivities

The signal level at the terminal of the RF coil, $s(t)$, was derived in the previous chapter, Eq. [2.36], and found to be

$$s(t) = -j\omega_0 \iiint_V B_{eff}(\vec{x}) M_{\perp}(\vec{x},0)^* e^{j\omega_0 t} e^{-\frac{t}{T_1}} e^{-\frac{t}{T_2}} d\vec{x}\,. \qquad [3.1]$$

The signal level is then amplified using a low noise pre-amplifier. The voltage presented at the terminal of the receiver, $r(t)$, is then simply Eq. [3.1] scaled by the gain, *G*, of the amplifier,

$$r(t) = Gs(t). \qquad [3.2]$$

For an array of coils, this is generalized as

$$r_i(t) = w_{ij} s_j(t), \qquad [3.3]$$

or in matrix notation,

$$\mathbf{r}(t) = \mathbf{W}\mathbf{s}(t), \qquad [3.4]$$

where **W** is a rectangular matrix containing weighting coefficients that describe how the coil elements are to be combined for each receiver channel. Converting the received signal from the time domain to the image domain using a Fourier transform operation,

$$\mathbf{i}(\vec{x}) = F\{\mathbf{r}(t)\}, \qquad [3.5]$$

the image from each coil is found to be

$$\mathbf{i}(\vec{x}) = \mathbf{W}\mathbf{B}(\vec{x}) M_{\perp}(\vec{x}). \qquad [3.6]$$

By inspection, the composite sensitivity, $\hat{B}$, associated with a receiver is

$$\hat{B}_i(\vec{x}) = \sum_j W_{ij} B_j(\vec{x}). \qquad [3.7]$$

Typically, each coil element has its own receiver and all the receivers were identical. This is a special case where **W** is a diagonal matrix where all the diagonal elements are equal to the pre-amplifier gain.

In cases where the number of coils is greater than the number of receiver channels, **W** will be a rectangular matrix where each row describes how the elements in the array will be combined to form composite coil sensitivities for each receiver channel. How these weighting coefficients are chosen determines the effectiveness of the array when a limited number of receiver channels are used.

Composite Resistance Matrix

Weighting the signal received from the array will also scale the noise for contribution from the array. Assuming the combination network is lossless and all elements are will isolated, the composite resistance matrix, $\hat{\mathbf{R}}$, for the array at the terminals of the receiver is

$$\widehat{\mathbf{R}} = \mathbf{W}\mathbf{R}\mathbf{W}^{\dagger}. \qquad [3.8]$$

If the combination network is not lossless, resistance needs to be added to the composite resistance matrix. Since the losses are within the combination network are assumed to be uncorrelated, it can be accounted for by adding the effective loss resistance for each channel, $\mathbf{R}'$, to the composite resistance matrix,

$$\widehat{\mathbf{R}} = \mathbf{W}\mathbf{R}\mathbf{W}^{\dagger} + \mathbf{R}'. \qquad [3.9]$$

Since these losses are uncorrelated, $\mathbf{R}'$ will be a diagonal matrix.

Coupling between channels within the combiner can be described by a coupling matrix, **C**. The coupling between elements will affect both the composite sensitivities as well as the composite resistance matrix of the array. Including coupling, the composite sensitivity is

$$\widehat{\mathbf{B}} = \mathbf{C}\mathbf{W}\mathbf{B} \qquad [3.10]$$

and the composite resistance matrix is

$$\widehat{\mathbf{R}} = \mathbf{C}\left(\mathbf{W}\mathbf{R}\mathbf{W}^{\dagger} + \mathbf{R}'\right)\mathbf{C}^{\dagger}. \qquad [3.11]$$

The relative SNR of the receiver using the composite sensitivity and composite resistance matrix is then found to be

$$\mathbf{SNR} = \frac{\widehat{\mathbf{B}}}{\sqrt{\widehat{\mathbf{R}}}}. \qquad [3.12]$$

For the case where each coil element has its own receiver and each receiver is identical, $\mathbf{W} = c\mathbf{I}$, and assuming there is no coupling within the combiner, the SNR is given by

$$SNR = \frac{\mathbf{W}\mathbf{B}}{\sqrt{\mathbf{W}\mathbf{R}\mathbf{W}^{\dagger}}} = \frac{c\mathbf{B}}{\sqrt{c^2\mathbf{R}}} = \frac{\mathbf{B}}{\sqrt{\mathbf{R}}}, \qquad [3.13]$$

and shows that the SNR is conserved through the lossless network.

Software Combination

Signal combination is implemented in software after all the coil signals have been acquired and digitized. The acquired signals are combined by multiplying them by a set of weighting coefficients, **w**, and then summing them. Mathematically, the combination of signals in the time, k-space, domain is described by

$$s(t) = \sum_{c=1}^{N_c} w_c a_c(t) \ . \qquad [3.14]$$

where N_c is the number of coils in the array, w_c is the weighting coefficient for the channel c, and $a_c(t)$ is the acquired signal from channel c.

It is also possible to combine the signals from the individual coils after the data has been processed and images reconstructed for each coil. Since the signal for each pixel is essentially independent from its neighbors, the combination can be performed on a pixel by pixel basis according to

$$I_{combined} = \sum_{c=1}^{N_c} w_c a_c \qquad [3.15]$$

where w_c is the weighting coefficient for channel c for that pixel, and a_c is the value of the pixel acquired from channel c.

How the images are combined depends on the choice of weighting coefficients and where the combination is implemented. If applied on the time domain signal prior to image reconstruction, the choice of weighting coefficients will affect the entire image. If applied on the image data, the weighting coefficients will affect individual pixels and can be different from pixel to pixel. The choice of weighting coefficients will affect the SNR of the pixel and is dependent upon the application. How to choose proper weighting coefficients will be discussed below.

Choosing Weighting Coefficients

The choice of weighting coefficients will determine the performance of the array in terms of SNR and/or g-factor. The coils in the array could be combined simply to

form a more complex, yet intuitive structure, that could easily be built. An example of this would be combined two loops to form a quadrature pair. A more useful application is to combine coils so that they produce optimal SNR within the image. Traditionally, using an array with a receiver for each element allows the image to be reconstructed with optimal SNR in every pixel. However, when the number of coils outnumbers the number of available receiver channels, optimizing for every pixel is no longer possible and it is better to choose weights that optimize SNR over regions instead. In this section, methods for determining the weighting coefficients are discussed.

Simple Combinations

A simple combination is defined as an array that has been formed from the combination of elements in a larger array and results in a reduction in the number of channels required to receive the signals. A simple example of a simple combination is a quadrature pair, Figure 5. The full array is composed of a loop and a saddle coil that are centered on one another. By combining the signals with a 90 degree combiner, $\mathbf{W} = \begin{bmatrix} 1 & j \end{bmatrix}$, the signals are combined onto one channel with a square root of two improvement in SNR at a point centered above the array (46).

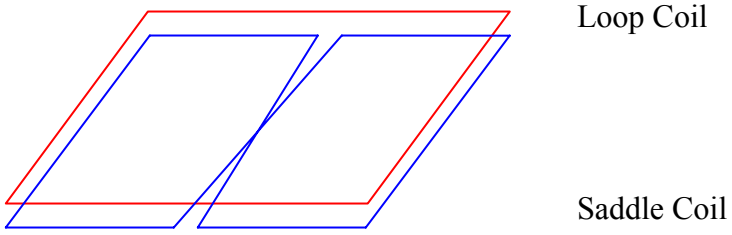
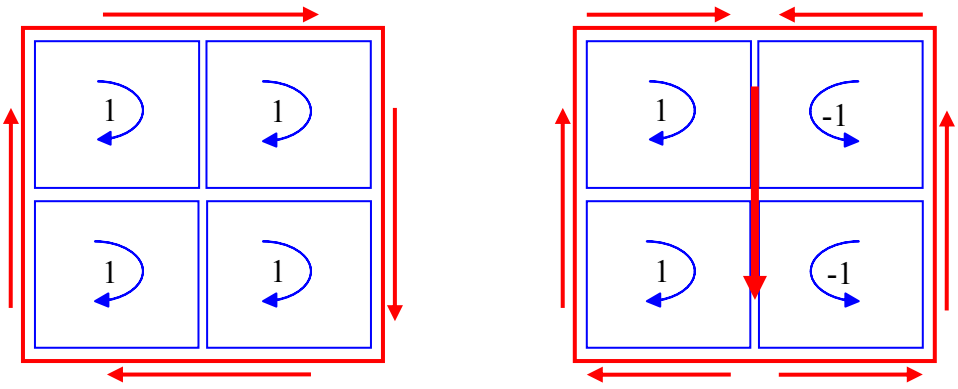


Figure 5. Traditional quadrature pair.

For a large array, the sensitivity of a channel is the superposition of the fields of the array elements weighted by the coefficients. For example, the simple case of all elements being identically weighted with $w = 1$, results in the combined receiver sensitivity being identical to a loop whose outer edge is the perimeter of the large array,

Figure 6a. By choosing the magnitude of the coil equal to one on each loops, the continuity of the current is enforced and the combined coil can be physically constructed with a single wire. The figure shows that when all loops are equal current, the inner conductors have oppositely directed currents and the resulting magnetic field due to these currents will cancel. The currents along the outside edge of the array are not canceled and will generate the magnetic field sensitivity of the large square loop. Similarly, a planar pair coil can be formed from the same array with by simply choosing a different set of coefficients, Figure 6b. The weight coefficients for a four element grid array are $\mathbf{W} = [1 \quad 1 \quad 1 \quad 1]$ to construct the loop coil and $\mathbf{W} = [1 \quad 1 \quad -1 \quad -1]$ to form the currents of a saddle coil.



(a) (b)

Figure 6. Combination of four small loops to form a large loop and planar pair.

If two receivers are available, it is possible to construct the quadrature pair from an array using weighting coefficients to combine the elements to effectively form the loop and saddle elements. For the four element grid array, these weighting coefficients are

$$\mathbf{W} = [1 \quad j]\begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ j & j & -j & -j \end{bmatrix} \qquad [3.16]$$

Obviously this can be easily extended to larger arrays with greater number of receivers.

An example of this would be a saddle train array. This array is composed of large complicated elements that can approximate a twelve element linear array with just four channels by

$$\mathbf{W} = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & -1 & -1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & 1 & -1 & -1 & -1 \end{bmatrix}. \qquad [3.17]$$

Since the array elements are physically constructed, the current must be continuous and therefore all the elements have the same amplitude. This limits the choice of coefficients to 0 or +/-1. A value of zero represents an inactive element and produces a gap in the element. The sign of the coefficient determines if the current in the loop is rotating clockwise or counterclockwise. The change in direction is achieved by inserting a crossover at this point in the coil.

For example, the first row of **W**, channel 1, are all equal to one meaning all elements are in phase and these add to form a large loop, Figure 7. In the second row, channel 2, the direction of the current reverse in the center of the coil and the elements add to form a saddle coil. Similarly, rows three and four combine to form saddle trains (13-15) with three and four lobes respectively.

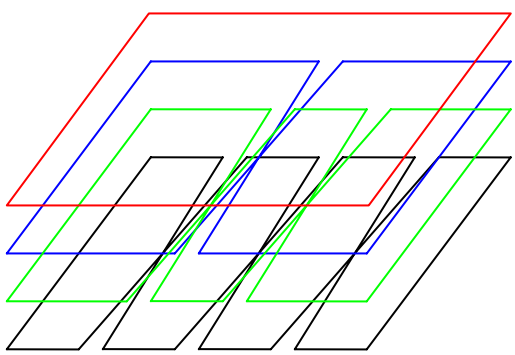


Figure 7. Four channel saddle train array.

Other geometries can also be constructed in this manner. A linear array of four overlapped surface coils is given by

$$\mathbf{W} = \begin{bmatrix} 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \end{bmatrix}. \qquad [3.18]$$

This array approximates a thirteen element linear array whose element width is equal to the width of the coil overlap. Another example is an array of planar pairs,

$$\mathbf{W} = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}, \qquad [3.19]$$

which contains information from an 8 element information array. Of course, these weighting coefficient matrices represent a reduction in rank and a loss of flexibility in combining the signals which results in a reduction in SNR performance.

Combining for Optimal SNR at a Point

It is not necessary to enforce continuity of current when combining arrays. By letting the weighting coefficients be equal to any complex value, the array can be used to form a complex current sheet over its surface. This current sheet would be extremely difficult, if not impossible, to construct as a single coil element. However, using the combiner, the array can be focused to give optimal SNR at points or over regions.

The derivation for the weighting coefficients to optimally combine an array to maximize SNR at a point has been published by Roemer and Wright(6,47). Basically, the derivation begins with the square of the SNR at a point,

$$SNR^2(\mathbf{x}) = \frac{\sum_i \sum_k n_i n_k B_i(\mathbf{x}) B_k(\mathbf{x}) \cos\left(\phi_i - \theta_i(\mathbf{x}) - \phi_k + \theta_k(\mathbf{x})\right)}{\sum_i \sum_k n_i n_k R_{ik} \cos(\phi_i - \phi_k)}, \qquad [3.20]$$

where n_i and ϕ_i are the magnitude and phase of the ith weighting coefficient, and B_i and

θ_i are the magnitude and phase of the effective, transverse, magnetic field of the ith array element. This equation is then differentiated with respect to the magnitude, n_i, and phase, ϕ_i, of the weighting coefficients. This results in a system of equations whose solution for the optimal coefficients is

$$\mathbf{W} = \lambda(\mathbf{x})\mathbf{B}^{\dagger}(\mathbf{x})\mathbf{R}^{-1} \qquad [3.21]$$

where λ is an arbitrary scaling constant. The scaling constant is generally chosen in order to produce a uniform noise level,

$$\lambda(\mathbf{x}) = \frac{1}{\sqrt{\mathbf{B}^{\dagger}(\mathbf{x})\mathbf{R}^{-1}\mathbf{B}(\mathbf{x})}}, \qquad [3.22]$$

or uniform signal level,

$$\lambda(\mathbf{x}) = \frac{1}{\mathbf{B}^{\dagger}(\mathbf{x})\mathbf{R}^{-1}\mathbf{B}(\mathbf{x})}, \qquad [3.23]$$

across the image.

An example of the point optimized weighting coefficients for a 9x9 12.375cm square grid array are shown for Figure 8. The weights are chosen to yield the optimal SNR at a point along the y-axis at 5cm above the plane of the array.

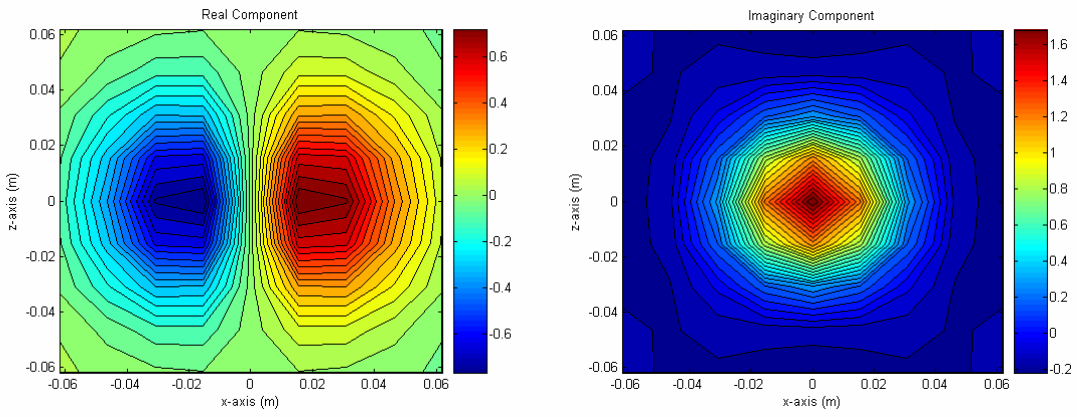


Figure 8. Weighting coefficients for a 9x9 point optimized grid array.

When multiple channels are available, each receiver can be assigned to give the optimal SNR at a different point in the imaging plane. The only decision required is which points to choose. These will depend on the application. Once the points are chosen for the optimal SNR, the weighting coefficients are given by the N-by-M matrix

$$\mathbf{W}_{n,:} = \frac{1}{\sqrt{\mathbf{B}(\mathbf{r}_n)^{\dagger}\,\mathbf{R}^{-1}\mathbf{B}(\mathbf{r}_n)}}\mathbf{B}(\mathbf{r}_n)^{\dagger}\,\mathbf{R}^{-1}. \qquad [3.24]$$

This set of coefficients combines the signals from M coils in a large array to N receiver channels so that the nth channel provides the optimal SNR at the points $\mathbf{r}_n$.

Combining for Optimal SNR over a Region

In some cases, particularly when the imaging plane is close to the surface of the array, the weighting coefficients for optimally combining the array for SNR at a point will produce a sensitivity that is highly focused about that point. This could produce nulls in the sensitivity pattern and significant loss in SNR away from the points. In order to solve this problem, Roemer's solution for optimal SNR at a point is extended to find the weighting coefficients to optimize SNR over a region or volume.

Starting with the square of the SNR at a point,

$$SNR^2(\mathbf{x}) = \frac{\sum_i \sum_k n_i n_k B_i(\mathbf{x}) B_k(\mathbf{x}) \cos\left(\phi_i - \theta_i(\mathbf{x}) - \phi_k + \theta_k(\mathbf{x})\right)}{\sum_i \sum_k n_i n_k R_{ik} \cos\left(\phi_i - \phi_k\right)} \qquad [3.25]$$

where n_i and ϕ_i are the magnitude and phase of the ith weighting coefficient, and B_i and θ_i are the magnitude and phase of the effective, transverse, magnetic field of the ith array element. This function is integrated over a region, Q, giving the magnitude of the SNR, λ, over the region,

$$\lambda = \iiint_Q SNR^2(\mathbf{x})\, d\mathbf{x}; \qquad [3.26]$$

or, after substituting Eq. [3.25],

$$\lambda = \frac{\iiint_Q \sum_i \sum_k n_i n_k B_i(\mathbf{x}) B_k(\mathbf{x}) \cos\left(\phi_i - \theta_i(\mathbf{x}) - \phi_k + \theta_k(\mathbf{x})\right) d\mathbf{x}}{\sum_i \sum_k n_i n_k R_{ik} \cos\left(\phi_i - \phi_k\right)}. \quad [3.27]$$

To find the weighting coefficients that maximize λ, the function, Eq. [3.27], is differentiated with respect to the magnitude and phase of the weighting coefficients, n_i and ϕ_i respectively, and setting the resulting equations equal to zero,

$$\frac{d\iiint_Q SNR^2(\mathbf{x}) d\mathbf{x}}{dn_i} = 0 \quad \frac{d\iiint_Q SNR^2(\mathbf{x}) d\mathbf{x}}{d\phi_i} = 0. \quad [3.28]$$

Evaluating these derivatives and letting

$$w_i = n_i e^{j\phi_i} \quad [3.29]$$

and

$$b_i(\mathbf{x}) = B_i(\mathbf{x}) e^{j\theta_i(\mathbf{x})} \quad [3.30]$$

yields a system of equations,

$$e^{j\phi_i} \sum_k w_k^* \iiint_Q b_i^* b_k d\mathbf{x} + e^{-j\phi_i} \sum_k w_k \iiint_Q b_i b_k^* d\mathbf{x} = \lambda \left(e^{j\phi_i} \sum_k w_k^* R_{ik} + e^{-j\phi_i} \sum_k w_k R_{ik} \right) \quad [3.31]$$

and

$$e^{j\phi_i} \sum_k w_k^* \iiint_Q b_i^* b_k d\mathbf{x} - e^{-j\phi_i} \sum_k w_k \iiint_Q b_i b_k^* d\mathbf{x} = \lambda \left(e^{j\phi_i} \sum_k w_k^* R_{ik} - e^{-j\phi_i} \sum_k w_k R_{ik} \right). \quad [3.32]$$

Taking the sum and difference of these equations, in order to solve them, leads to

$$\sum_k w_k^* \iiint_Q b_i^*(\mathbf{x}) b_k(\mathbf{x}) d\mathbf{x} = \lambda \sum_k w_k^* R_{ik} \quad [3.33]$$

and

$$\sum_k w_k \iiint_Q b_i(\mathbf{x}) b_k^*(\mathbf{x}) d\mathbf{x} = \lambda \sum_k w_k R_{ik}. \quad [3.34]$$

Eqs. [3.33] and [3.34] are complex conjugates of each other and the resulting system of

equations can be succinctly expressed as

$$\mathbf{Bw} = \lambda \mathbf{Rw} \qquad [3.35]$$

where **B**, a mutual sensitivity matrix, is the correlation of the coil sensitivities over the region Q,

$$B_{ik} = \iiint_Q b_i(\mathbf{x}) b_k^*(\mathbf{x}) d\mathbf{x} . \qquad [3.36]$$

Bringing the mutual resistance matrix, **R**, to the left hand side of Eq. [3.35],

$$\mathbf{R}^{-1}\mathbf{Bw} = \lambda \mathbf{w} , \qquad [3.37]$$

puts it in the form of an eigenvalue problem. Since the eigenvalues, λ, also quantify the SNR over the region, Eq. [3.26], the eigenvector associated with the largest eigenvalue contains the weighting coefficients that maximize the SNR over the region,

$$[\mathbf{w}, \lambda] = \text{eig}\left(\mathbf{R}^{-1}\mathbf{B}\right)\Big|_{\lambda=\lambda_{\max}} . \qquad [3.38]$$

An example of the of this method is shown in Figure 9. The figure shows the real and imaginary weighting coefficients necessary to combine a nine-by-nine element planar grid array to optimize SNR over a 10cm by 10cm FOV at a depth of 5cm above the plane of the array.

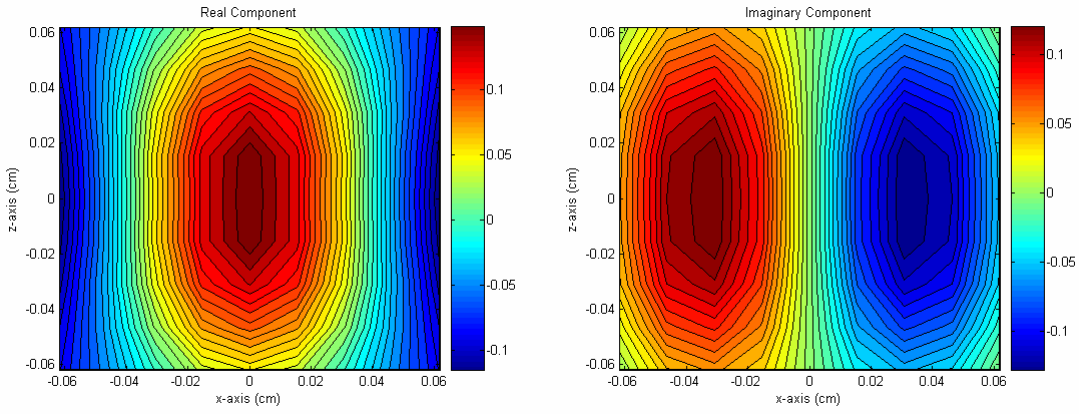


Figure 9. Region optimized weighting coefficients for a 9x9 grid array.

When multiple receivers are available, each channel can be assigned a different region of the image. Usually, the thickness region would be chosen to be on the order of the slice thickness, but for three dimensional imaging protocols, the region can be set to cover the entire imaging volume.

Combining for Optimal G-Factor

Previously, in Chapter II, the g-factor was defined as well as the conditions under which the g-factor could be minimized over the image and set equal to one. The simplest situation meeting these conditions is where the signal from each aliased pixel in the image is detected by only one receiver. This is implemented by having a receiver channel sensitive over a region of the image and having zero sensitivity outside this region. A simple example of this is assigning each channel the sensitivity pattern of a *rect* function. The ideal profile of the sensitivity for each channel is shown in Figure 10. The profiles are spaced equally in the phase encode direction since this is the direction in which the k-space is undersampled and aliasing occurs. Care must be taken in defining the forcing function used to fit the desired channel sensitivities. If the width of the combined sensitivity of the channel is larger than the width of the aliased FOV of the reduced image data set, there will be an ambiguity within the aliased image of the location of the pixels and image reconstruction will be impossible.

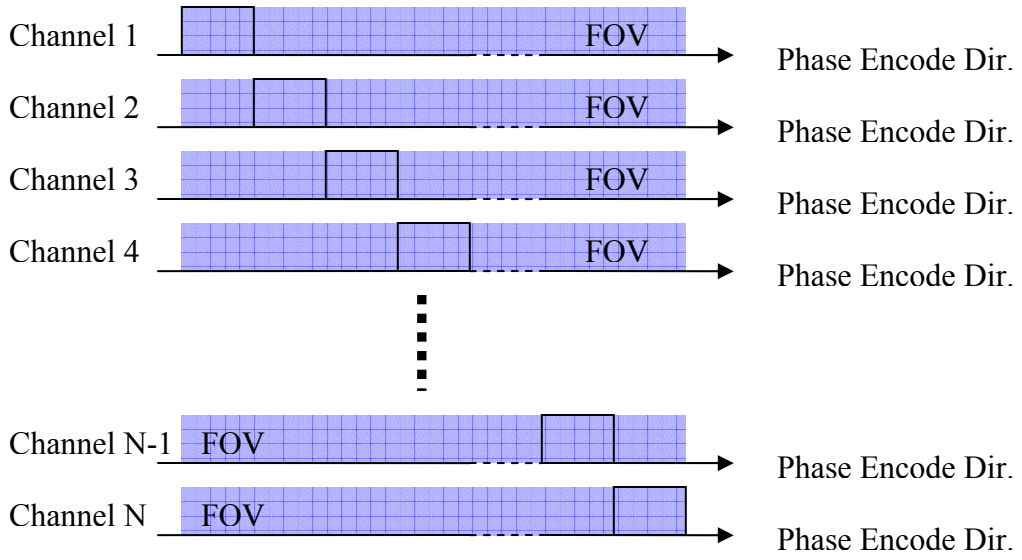


Figure 10. Ideal *Rect* receiver sensitivity profiles.

In order to find the weighing coefficients to combine the array to fit the desired forcing functions, a system of equations is set up,

$$\mathbf{Bw} = \mathbf{F}, \qquad [3.39]$$

to solve for the coefficients, **w**. The matrix **B** contains the sensitivities of the array elements at a number of test points in the FOV and **F** contains the values of the forcing functions at the test points. At a minimum, the number of test points used to fit the channel sensitivity needs to be equal to the number of coil elements in the array. This will find coefficients that force the channel sensitivities at the test points. However, away from the test points, the sensitivities are not constrained and may take any value.

An example of this, using a 64 element loop array, is shown in **Figure 11** where the channel sensitivities have been fitted to thirty-two test points along four rows for a total of 128 points. Four equal spaced rows, parallel to the phase encode direction, are used to enforce uniformity of coverage in the readout direction. Since a linear array was used for these simulation, four rows were sufficient. However, if a grid array were being used, many more rows might be necessary. The plot shows the sensitivities along the

center of the FOV at each pixel in a 256 point image. The values of the forcing function at the test points are either zero or one depending on if the channel is sensitive at this point. In between the test points, there is significant error in the profiles.

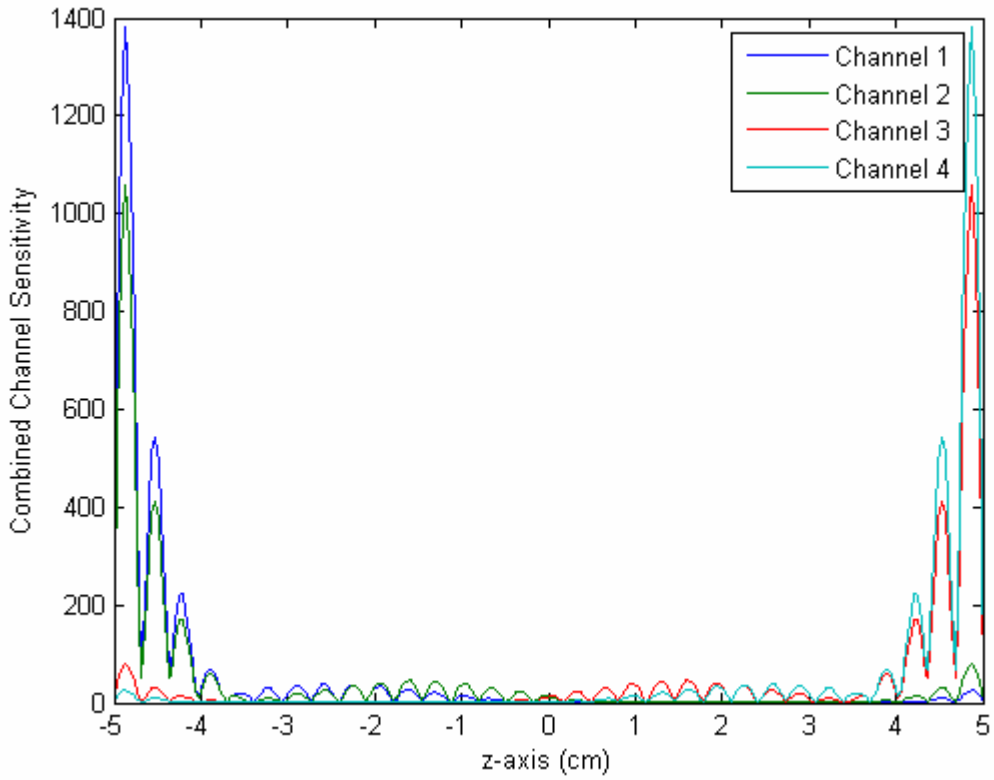


Figure 11. Four channel sensitivity profile fitted at thirty-two test points on four rows.

In order to better fit the desired sensitivity over the entire region, more points should be used. Figure 12 plots the sensitivity profiles when a 256 test points, on four rows, are used. The position of the test points coincide with center of the pixels in the final image. The fitted profiles are now forced to equal the fitting fuctions at each pixel and now closely approximate the desired *rect* functions. It is possible, however, to use too many test points, Figure 13. When this occurs, numerical instabilities in the algorithm become significant and the fitted profiles begin diverge from the desired result.

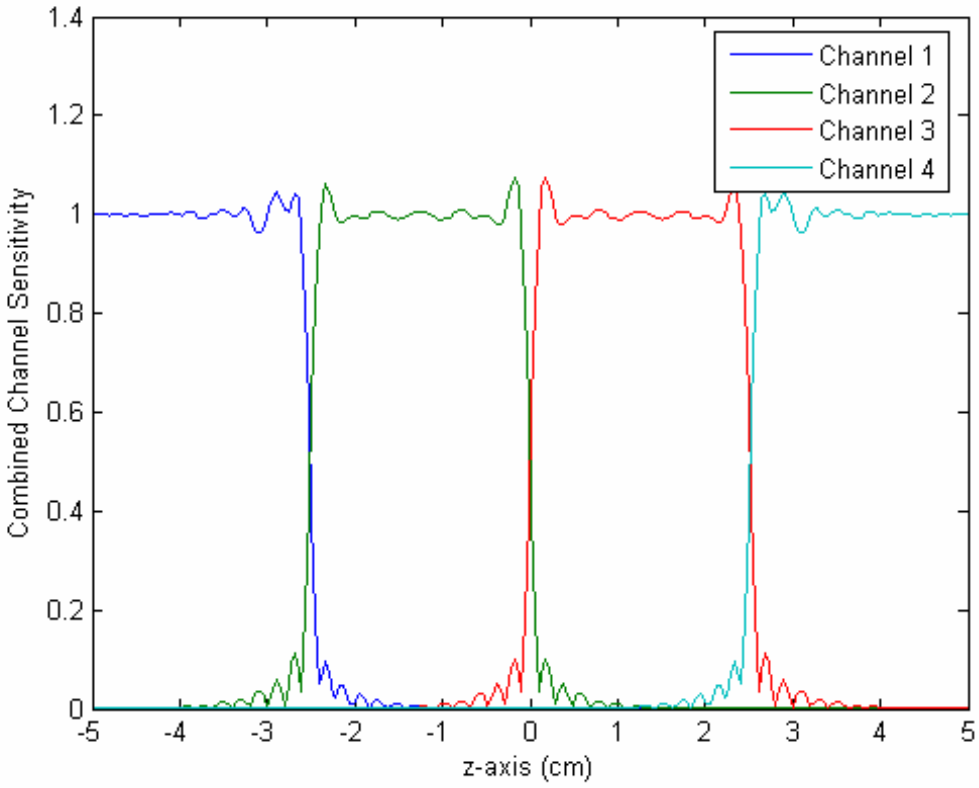


Figure 12. Four channel sensitivity profile fitted at 256 test points on four rows.

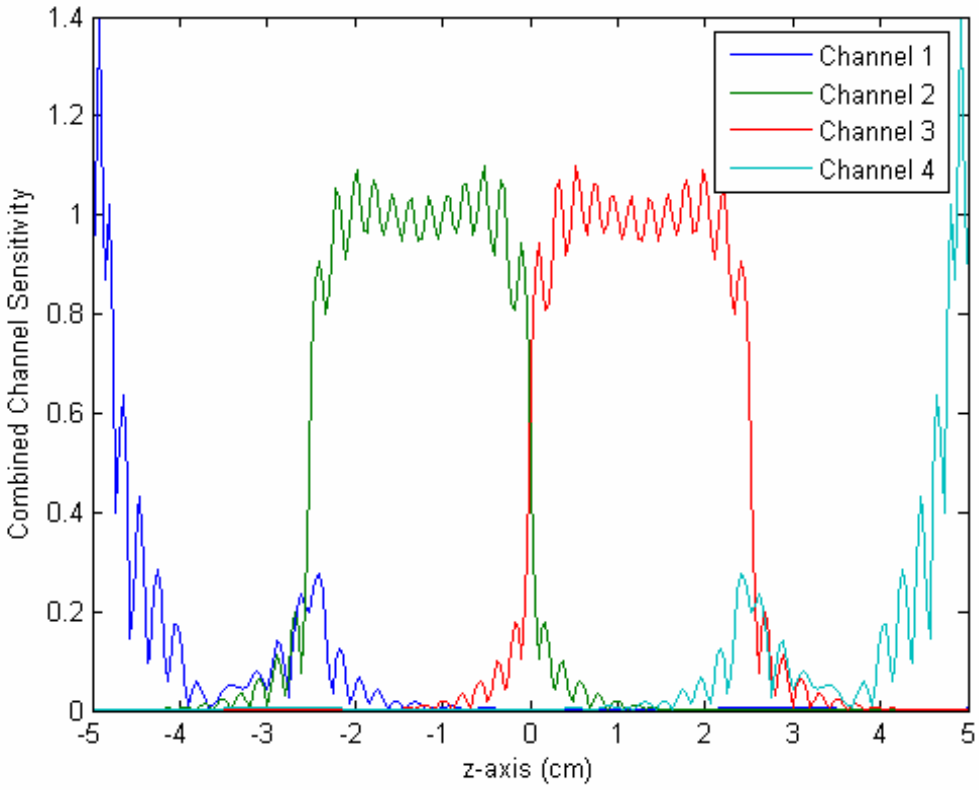


Figure 13. Four channel sensitivity profile fitted at 1024 test points on four rows.

In these examples, the fitting function was chosen to be a simple *rect* function. It is possible that better functions for optimizing the g-factor as well as the overall SNR. In Chapter V, a phase ramp is applied to the *rect* function, with each channel having a different slope, to improve the g-factor of the reconstructed images. Functions that improve upon these results are left for future work.

CHAPTER IV

IMPLEMENTATION

The models implemented to study the combination of arrays for SENSE imaging were implemented in several different modules. Operations required significant computing resources were implemented in C++, while operations that were generally matrix operations and essentially linear algebra were implemented in Matlab®. Display of the data and results were generally done in Matlab® as well using binary files imported from the C++ programs.

C++

The C++ algorithms were developed using Microsoft Visual C++®. C++ was used for algorithms that required significant amounts of memory, processing time, or needed a graphical user interface. The algorithms needed for this dissertation that are written in C++ are one for computing mutual resistance, RMat.exe, and an implementation of a genetic algorithm for combining arrays while enforcing continuity of current, MV_SENSE.exe.

Computing Resistance

Evaluating the mutual resistance matrices due to sample loading was implemented in C++ due to the large number of three dimensional integrals that needed to be computed. The integration method implemented is a recursive three dimensional version of Simpon's Rule which iterates until either a minimum degree of convergence is attained or a maximum number of recursions is reached. The program requires several inputs, Table 1, and outputs the resulting resistance matrix to a text file that can then be imported into Matlab® or viewed with a text editor.

Table 1. Inputs for computing R Matrix

INPUT	DEFINITION
Rows	# of rows in the array running perpendicular to z-axis
Columns	# of columns in the array running parallel to z-axis.
Length	Length of the array in meters parallel to z-axis
Width	Width of the array in meters perpendicular to z-axis
Plane	Angle in degrees from x-axis of the array plane
Plane Center	Coordinate of center of array in meters
Sample Center	Coordinate of center of sample volume in meters
Sample Dimensions	Vector with length, width, and height of sample in meters

The computation assumes that the array elements are rectangular loops with dimensions

$$l = \frac{L}{N} \qquad [4.1]$$

and

$$w = \frac{W}{M}\,. \qquad [4.2]$$

The analytical quasi-static solution for the magnetic vector potential for a straight wire with uniform current is used to eliminate one integration step,

$$MVP\_WIRE(\mathbf{x},\mathbf{s},\mathbf{e}) = \frac{\mu}{4\pi}\int_{\mathbf{s}}^{\mathbf{e}} \frac{1}{|\mathbf{x}-\mathbf{x}'|}\,\mathbf{dx}'\,. \qquad [4.3]$$

The inputs for this function are the observation point, **x**, the starting point of the straight conductor, **s**, and its ending point, **e**. The magnetic vector potential for the loop is then found by summing the contributions of the four wires in the loop,

$$MVP\_LOOP(\mathbf{x},\mathbf{c},l,w) = \sum_{i=1}^{4} MVP\_WIRE(\mathbf{x},\mathbf{s}_i,\mathbf{e}_i) \qquad [4.4]$$

where **c** is the center of the loop and l and w are the length and width of the loop respectively. The resistance matrix is then filled using

$$R(i,j,\mathbf{sc},\mathbf{sd},\omega,\sigma)=\sigma\omega^2\iiint_V MVP\_COIL(\mathbf{x},\mathbf{c}_i,l,w)\bullet MVP\_COIL(\mathbf{x},\mathbf{c}_i,l,w)\,\mathbf{dx} \quad [4.5]$$

where **sc** and **sd** are the vectors containing the sample center and sample dimensions respectively.

Genetic Algorithm

In order to investigate the advantages of constructing "global" arrays and "local" arrays a program was written that could evaluate and optimize an array geometry for SNR and/or g-factor. This program was written in C++ and has a windows user interface to give maximum flexibility, Figure 14. Displays of coil geometry, SNR maps, g-factor maps are shown for an array. The program was written using object oriented code to allow flexibility and scalability and includes multiple threads so that computationally intensive tasks can run in the background.

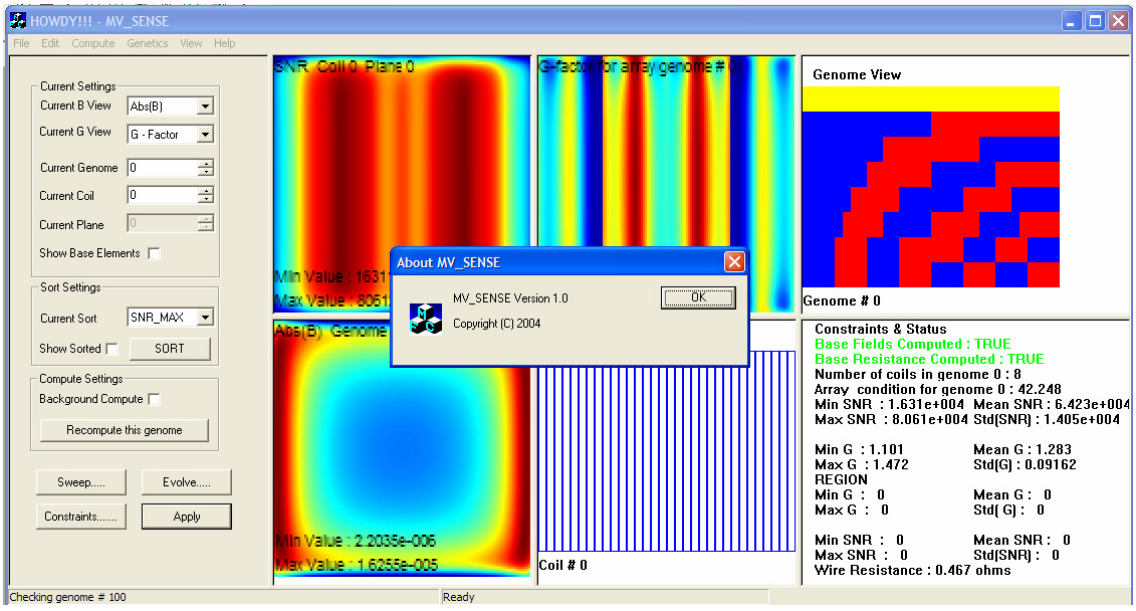


Figure 14. MV_SENSE display.

The programming model used for this application, MVSENSE, is the Document/View architecture. In this architecture, all the data necessary to recreate a project is stored in the objects derived from the document class. All the functions necessary to view and modify the data are handled by objects derived from the view class. A block diagram of the base class interaction is shown in Figure 15. A list of the major objects are listed in Table 2 and a description of each is below. The program was written using Visual C++® and many of the classes are derived from the MFC++® classes.

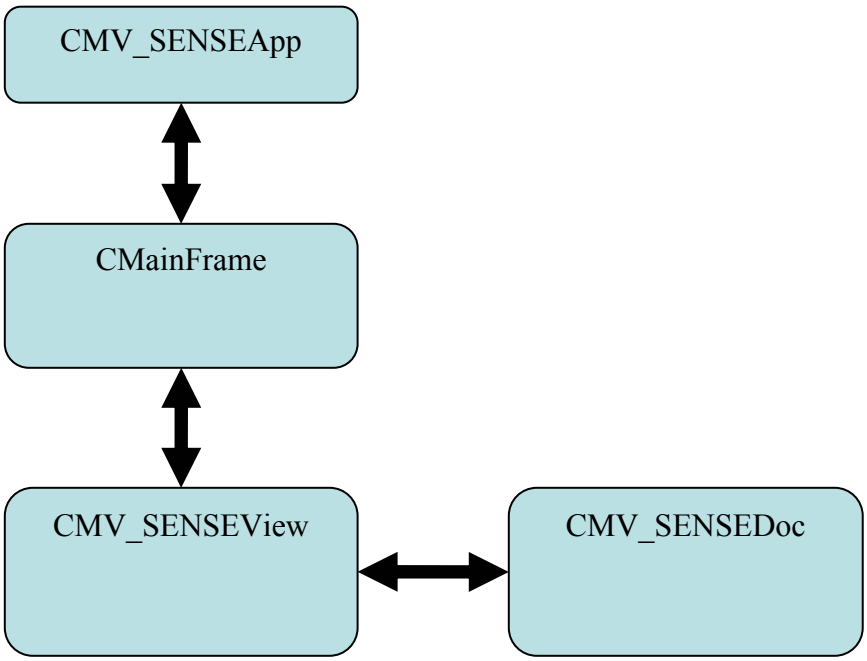


Figure 15. MV_SENSE block diagram.

Table 2. Class descriptions for MV_SENSE

APPLICATION CLASSES	DESCRIPTION
CMV_SENSEApp	Main Application Object
CMV_SENSEDoc	Base Document Class for containing application data
CMV_SENSEView	Base View class for displaying document data
CMV_SENSEGeoView	Displays array element geometry in a subframe
CMV_SENSECtrlView	Provides buttons and list boxes to control views of data
CMV_SENSEConstView	Displays application constants
CMV_SENSEGenView	Displays the overall array geometry
CMV_SENSEMapView	Provides common support for updating and displaying bitmaps
CMV_SENSE_BMapView	Displays sensitivity information for a coil element
CMV_SENSE_GMapView	Displays g-factor for an array
CMV_SENSE_SNRMapView	Displays the SNR map for an array
DATA CLASSES	
CConstraints	Contains user variables
CArrayGenome	Contains data necessary to construct an array
DIALOG CLASSES	
CConstraintsDialog	Provides dialog box for editng constraints
CEvolveDialog	Provides dialog box for controlling genetic algorithm
CSweepDialog	Provides dialog box for sweeping constraints

CMV_SENSEApp

The application class is derived from CWinApp and provides the API interface with the Windows® operating system. This creates the main user interface thread, UIThread, which is registered with the operating system. CMV_SENSEApp is the owner of the main windows frame and provides the main message loop for the application. During initialization, the Document/View template is created and attached

to this process, the main windows is subdivided into seven panes, a view class is attached to each pane, the status bar is initialized, and finally, the message loop is started.

CMV_SENSEDoc

The document class not only contains all the data in a class, but also contains the functions to operate on that data. The object contains a container class array that holds the descriptions for an entire population of coil arrays. The descriptions themselves are contained in the class CArrayGenome. The class also contains the magnetic field sensitivities and resistance matrix of the basis array as well as the functions for computing them.

The document class provides the support for saving, loading, importing, and exporting the data contained in the document into several formats. The entire document can some saved or loaded using streams. Specific data, such as coil sensitivities or a resistance matrix that takes time to evaluate, can be exported into a binary or text file for use by another program.

CMV_SENSEView

The view class is responsible for providing "views" of the data contained in the document. In this application, CMV_SENSEView just provides basic view functionality and acts as a foundation for the classes CMV_SENSEGeoView, CMV_SENSECtrlView, CMV_SENSEConstView, and CMV_SENSEMapView.

CMV_SENSEMapView

This class, derived from CMV_SENSEView, provides the basic support for rendering a bitmap to a window pane. The object takes a pointer to a two dimensional array of integer values, between 0 and 255, and generates a bitmap using the selected color map; grayscale or color. String variables are made available to superimpose text upon the bitmap. In order to increase display speed, the bitmaps are first created in a virtual display in memory and then copied to the video memory. Using a virtual display

also enables the user to copy an image from the window pane to the clipboard to be pasted into another windows application. This object serves as the base class for the objects used to create map from specific data; CMV_SENSE_BMapView, CMV_SENSE_GMapView, and CMV_SENSE_SNRMapView.

CMV_SENSE_BMapView

This class displays the magnetic field sensitivity of the selected coil element. The element can be selected using the up/down arrow keys or the mouse wheel either in this pane or in the CMV_SENSEGenView pane. The map shown is either of Bx, By, the phase of the element sensitivity, or its magnitude. The output of this object is shown in Figure 16 and is circled.

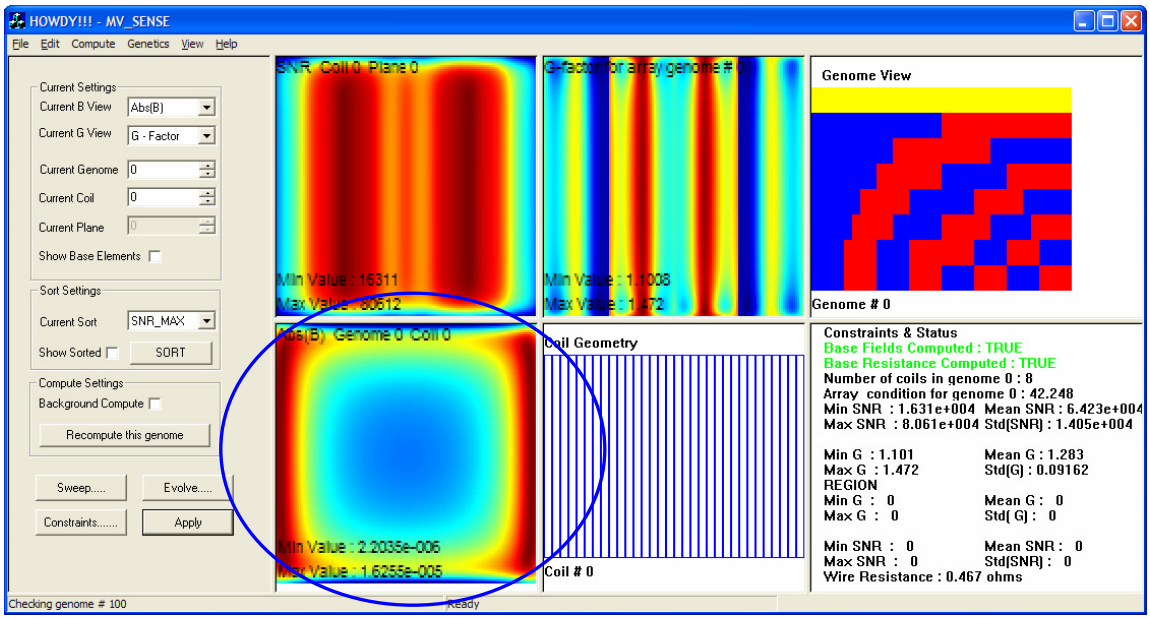


Figure 16. Window pane for element sensitivity, CMV_SENSE_BMapView.

CMV_SENSE_SNRMapView

This object evaluates the combined SNR map for the selected array and then scales it to be used by CMV_SENSEMapView. The SNR is combined using

$$SNR_{\max} = \sqrt{\mathbf{S}^{\dagger}\mathbf{R}^{-1}\mathbf{S}}\,. \qquad [4.6]$$

The array to be viewed is selected using the left/right arrow keys in the CMV_SENSE_GenView pane. The object also determines statistical information about the SNR map which is displayed in the CMV_SENSE_ConstView pane. The information computed is the maximum value, minimum value, mean, and standard deviation of the SNR map. An example is shown in Figure 17 and is circled.

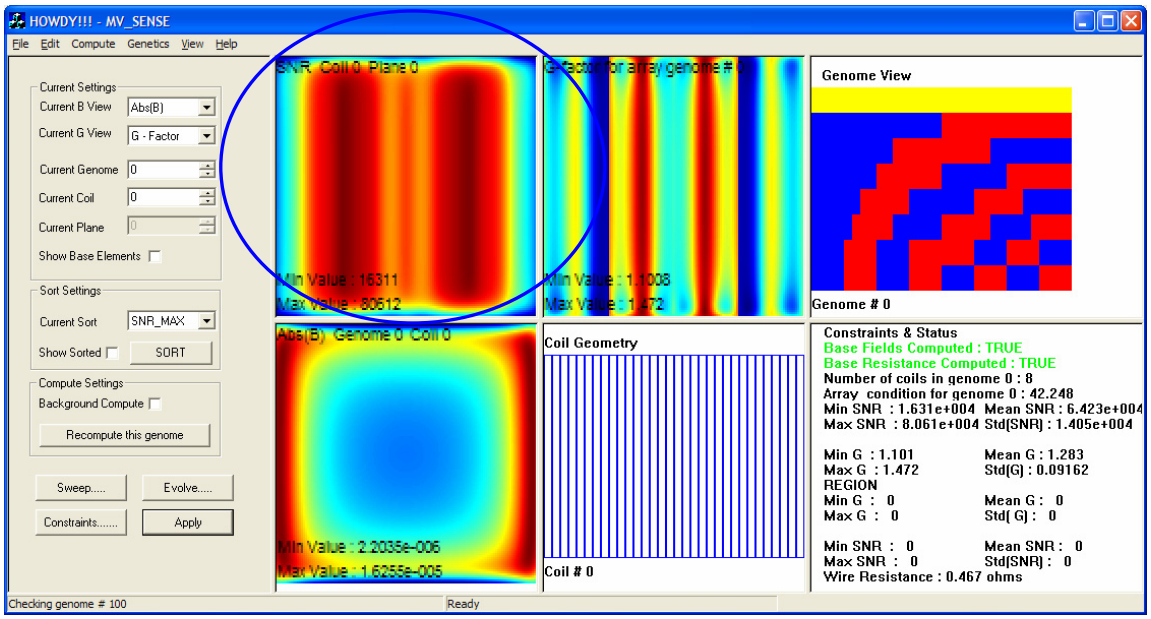


Figure 17. SNR map of an array genome.

CMV_SENSE_GMapView

The g-factor of the selected array is displayed using the object CMV_SENSE_GMapview. Using the reduction factor setting in CConstraints, the g-factor for the array is calculated according to

$$g_i = \sqrt{\left(\mathbf{S}^{\dagger}\mathbf{R}^{-1}\mathbf{S}\right)^{-1}_{i,i} \cdot \left(\mathbf{S}^{\dagger}\mathbf{R}^{-1}\mathbf{S}\right)_{i,i}}\,. \qquad [4.7]$$

The resulting matrix is then scaled and passed to CMV_SENSE_MapView to be

rendered into a bitmap. The object also finds the maximum and minimum values of g-factor as well as the mean and standard deviation. These values are displayed in the MV_SENSE_ConstView pane. An example is shown in Figure 18 and CMV_SENSE_GMapView pane is circled.

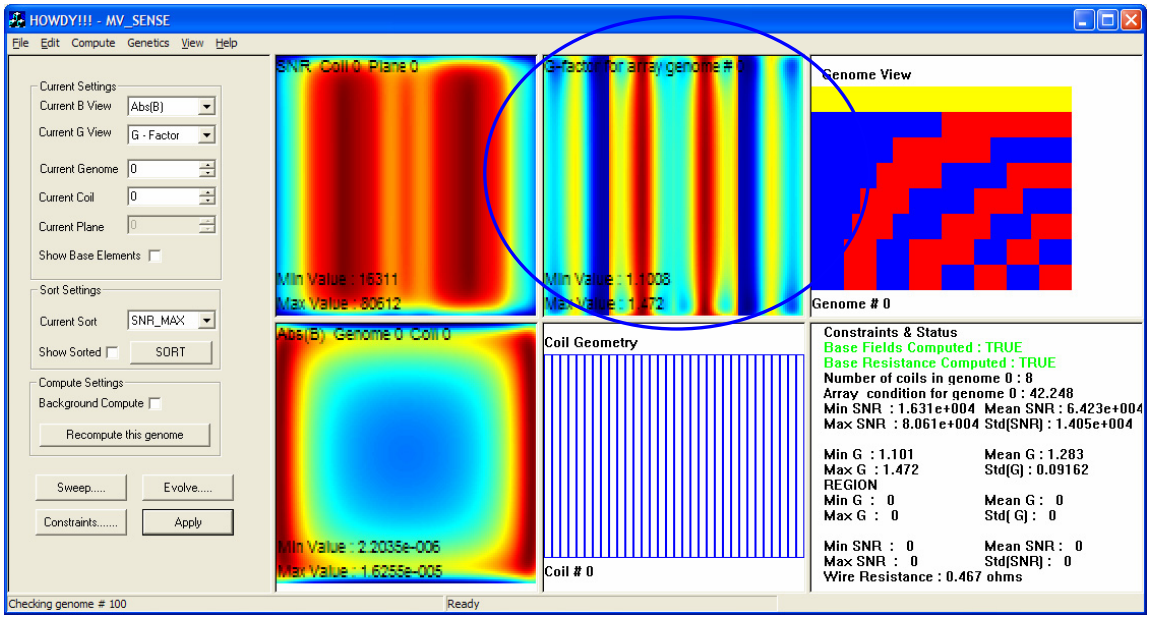


Figure 18. G-factor map display.

CMV_SENSEGeneView

The entire array topology is expressed graphically using this object. The reason for calling it gene view is that for the genetic algorithm portion of the program, an array is expressed as a genome. Each row in this display represents a different coil element or gene. Each gene is made up of alleles, in this case a pair of bits, that correspond to a basis function. The alleles may have one of three states; one, negative one, or zero. Figure 19 shows the program display with the pane corresponding to CMV_SENSEGeneView circled.

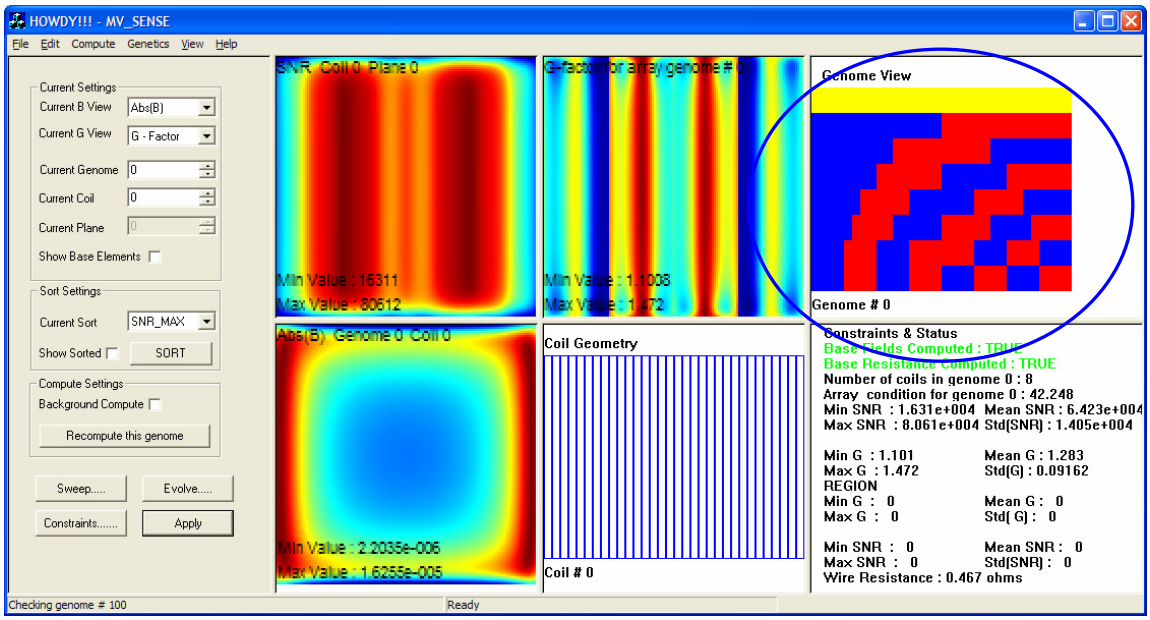


Figure 19. Display of array genome.

In this example, the pane has eight rows that correspond to eight array elements. Each row contains sixteen alleles. The alleles represent the direction of current in the basis element corresponding to this location. The colors represent the state of the allele. Blue represents current flowing in the clockwise direction in this basis element. Red means current is flowing counter-clockwise. White, not shown in the example, means no current is present in that basis element. The yellow bar means that this row, coil, has been selected for display by the CMV_SENSE_BMapView and CMV_SENSEGeoView.

The reason this bar is yellow is that its pixel values have been masked using an exclusive-or operation. By pressing the up or down arrow keys or using the mouse wheel while the cursor is in this pay, the mask will shift and the row will return to its proper color, in this case blue. Using the left or right arrow keys navigates the display between different array genomes. The program is initialized to handle one hundred different arrays.

CMV_SENSE_GeoView

The user interface for programming and manipulating an array element is controlled by the CMV_SENSE_GeoView pane, Figure 20. This pane provides a top down view onto the plane of an array element. Each basis element is shown in this view in one of three colors; blue, red, or white. Blue represents that the current is flowing in the clockwise direction in that basis loop. Red means the current is flowing counter clockwise. White, which is also the background color, means no current flows in that basis. In the example, all the elements appear in blue which, when added together, form a surface coil with dimensions equal to the size of the basis array.

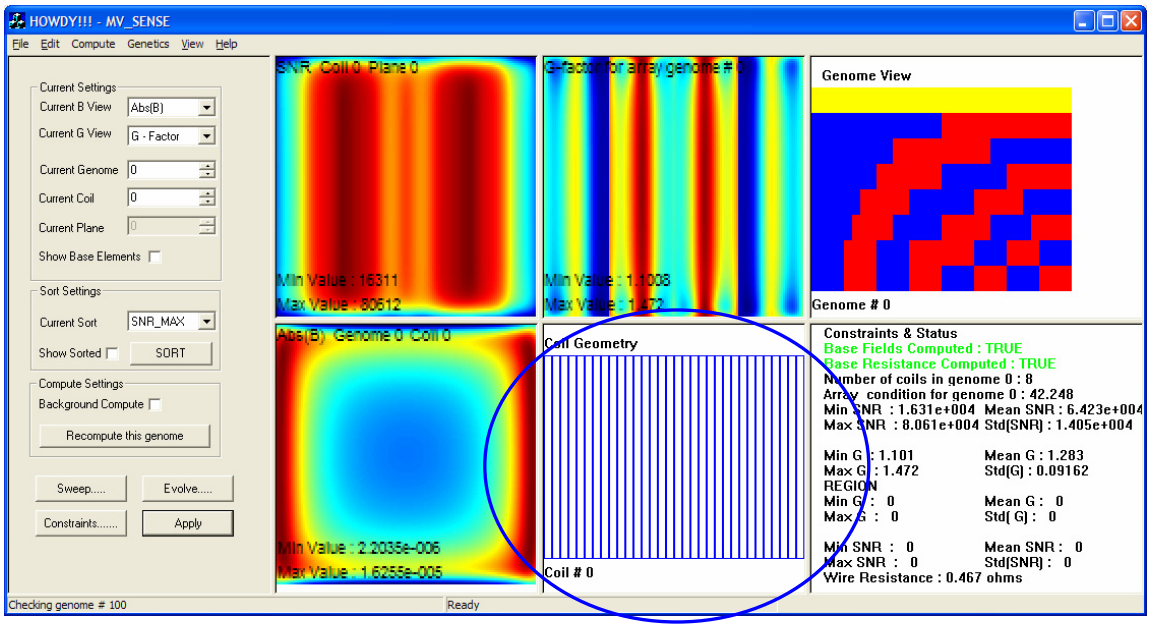


Figure 20. Display of array element geometry.

The state of a basis function is changed by clicking on it with the left mouse button. It takes four mouse clicks to return the basis function it its original state, due to two bits being used to encode the state. The changes are updated after the "Recompute this Genome" button is clicked in the CMV_SENSE_CtrlView pane.

CMV_SENSE_CtrlView

This view, Figure 21, is the largest in the application and contains list boxes, radio buttons, and pushbuttons by which the application is controlled. The view contains four regions: Current settings, Sort settings, Compute Settings, and four pushbuttons.

The current settings section controls the current displays. The first list box in this section, "Current B View", controls the how the magnetic field sensitivity is displayed. The available options are "Bx", "By", "angle(B)", and "abs(B)". Choosing "Bx" displays the $\hat{\mathbf{x}}$-directed component of the magnetic while "By" displays the $\hat{\mathbf{y}}$-directed component. Angle(B) shows the phase of the magnetic field, $\angle B_{eff}$, over the field of view and abs(B) gives the magnitude, $\left|B_{eff}\right|$.

The second list box, "Current G View", controls what is displayed in the g-factor display. The settings available in this list box are g-factor, variance, and inverse variance. The default is to simply display the g-factor map for the array. It is sometimes useful to display the components of the g-factor. The variance for the array is given by

$$V_i = \left(\mathbf{S}^{\dagger}\mathbf{R}^{-1}\mathbf{S}\right)_{i,i} \qquad [4.8]$$

and is displayed when the "variance" setting is chosen. Similarly, the inverse variance is given by

$$V_i^{-1} = \left(\left(\mathbf{S}^{\dagger}\mathbf{R}^{-1}\mathbf{S}\right)^{-1}\right)_{i,i} \qquad [4.9]$$

and is shown when the "inverse variance" setting is selected.

The third and fourth boxes, "Current Genome" and "Current Coil" control which array genome and which array element, respectively, are being viewed. The allowable range for the "Current Genome" control is between 0 and 99. Likewise, the range for the "Current Coil" setting is between 0 and the maximum number of coil elements.

The checkbox at the bottom of this section controls where the array genomes or the basis elements are displayed. When checked, the SNR Map pane and B Map pane will show the maps corresponding to the selected element. The G Map pane as well as the "Current Genome" control are disabled since they have no meaning in this context.

The “Current Coil” control now has a range of zero to the number of basis elements.

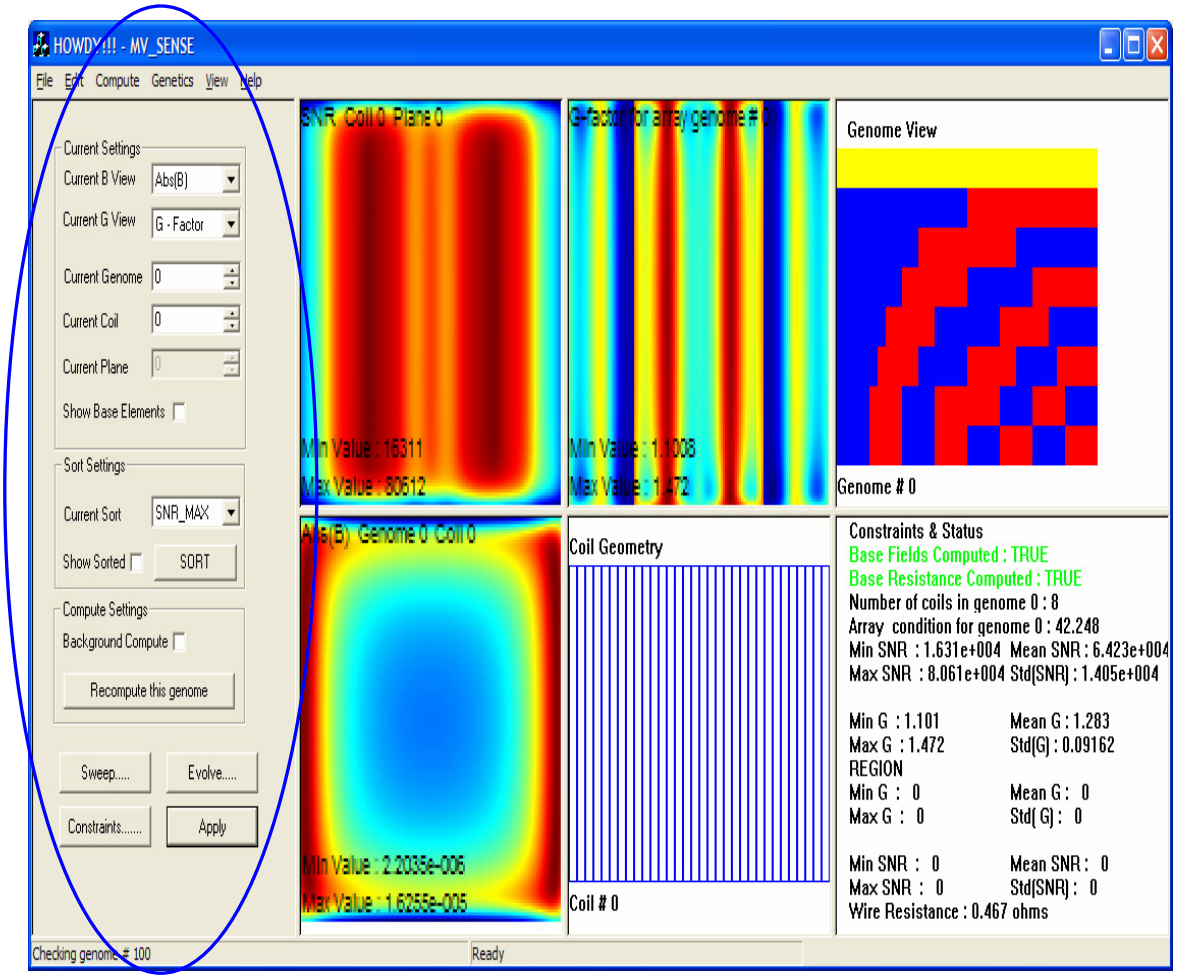


Figure 21. Control view pane.

The second section, “Sort Settings”, controls how and if the array genomes in the population are sorted. The list box selects the current sort parameter. The parameters available are the maximum SNR, “SNR_MAX”, the minimum SNR, “SNR_MIN”, mean SNR, “SNR_MEAN”, the standard deviation of the SNR, “SNR_STD”, the maximum g-factor, “G_MAX”, the minimum g-factor, “G_MIN”, the mean g-factor, “G_MEAN”, the standard deviation of the g-factor “G_STD”, and unsorted, “NONE”. The sort is executed by pressing the “SORT” pushbutton. The results of the sort are shown if the if the “Show Sorted” radio button is checked, otherwise the arrays are in

their initial order.

The third section, “Compute Settings”, contains a radio button for enabling background computation of the array genomes and a pushbutton, “Recompute this genome,” for immediately computing the displayed genome. The background computation is useful when a new population of genomes is created. Before a sort can be started, all the array genomes have to be evaluated and filled. This can take time and is best done in the background so that the application doesn’t appear “frozen”.

The last section contains four pushbuttons that launch dialog boxes and apply the current settings. The default button is “Apply” and updates the display panes to represent the values set in the “Current Settings” controls. Since it is the default, it can be activated either by clicking on it or pushing the enter key. The other three buttons launch dialog boxes. The button “Constraints…” creates a modal instance of CConstraintsDialog. Similarly, the buttons “Sweep…” and “Evolve…” launch modal instances of CSweepDialog and CEvolveDialog respectively.

CConstraints

This object is a custom data object that contains the global variables for all the arrays. A list of these global variables and their description is given in Table 3. All dimensions of length are defined to have units of meters. The object provides functionality to serialize its data and to read or write the variables to either streams or files in both text and binary formats. The object also includes basic logical operations to determine equality of two CConstraints objects and to set one object equal to another.

Table 3. Variable descriptions for CConstraints

Variable Name	Type	Description
left_edge	double	left edge position of basis array in x-direction
right_edge	double	right edge position of basis array in x-direction
length	double	length of basis array in z direction
number_of_planes	integer	number of planes in which a basis elements exist
planes	Vec<double>	a dynamic vector containing the y coordinate of the planes
FOV_x	double	width of imaging plane in x-direction
FOV_z	double	width of imaging plane in z-direction
imaging_depth	double	distance of imaging plane from array plane
resolution_x	integer	number of pixels in the x-direction
resolution_z	integer	number of pixels in the z-direction
sample_dimension	double[3]	vector containing length, width, and depth of sample
sample_center	double[3]	vector containing the center point of the sample
sample_conductivity	double	conductivity of the sample in Siemens.
frequency	double	operating frequency in Hertz
wire conductivity	double	conductivity of the wire in Siemens
wire_radius	double	effective radius of the conductor
reduction_factor	integer	reduction factor for SENSE imaging
max_elements	integer	the number of elements in the basis array

CArrayGenome

The information for describing an array is contained in this object. The object also contains statistical information for comparison. All the variables in this object are declared as private and can only be accessed through member functions. This was done to prevent accidental modification by the Sweep and Evolve algorithms. The member variables and a brief description are given in Table 4. SNR and g-factor statistics may be evaluated over the entire FOV or a specified region.

The information on how basis functions are composed to create an array element is stored in coil_elements. This is a two-dimensional array of 8 bit characters. The first dimension corresponds to the number of elements in the array. The second dimension is equal to the number of basis elements in the array divided by four since only two bits are required to set the state of the basis element for the coil. The plane of basis elements used to form the array is specified by plane. All basis elements used to generate an array element are from the same plane.

Table 4. Member variables of CArrayGenome

Variable Name	Type	Description
number_of_coils	integer	number of coils in the array
coil_elements	unsigned char**	"genes" describing the array elements
element_size	unsigned int*	number elements in an array element
plane	unsigned char*	plane a coil element is in
constraints	CConstraints	contains the variables for which the field maps were computed
Mat_Resistance	Mat<double>	matrix containing the array mutual resistance
Mat_G_Map	Mat<double>	matrix containing the current g-factor map
Mat_SNR_Map	Mat<double>	matrix containing the current SNR map

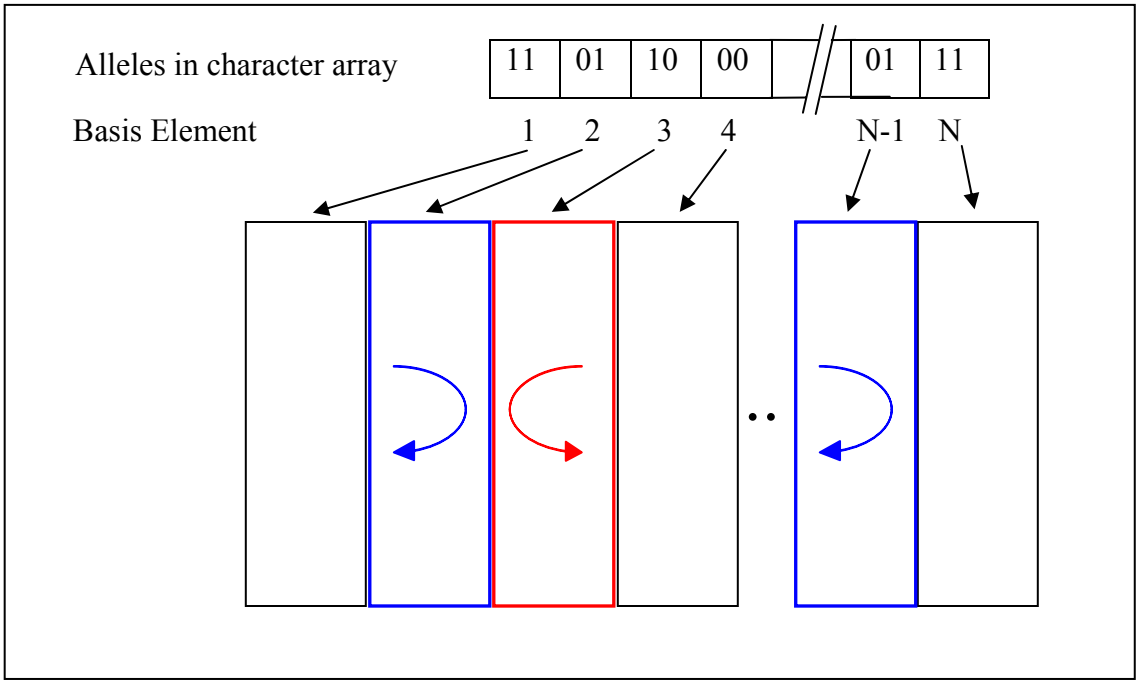


Figure 22. Element encoding diagram.

The coil elements are encoded using two bit characters which encode four possible states, Figure 22. Since only three states are required to specify the current in a loop, the fourth state is redundant and maps to zero current. The codes 01 and 10 were used to specify clockwise and counter clockwise currents so that exclusive or operations could be used during the "mating" portion of the genetic algorithm. The use of the bit array not only minimizes the memory requirements necessary to describe an array, but also removes a lot of "garbage" from the genome. If an entire integer were used to describe a genome, with values of 1, 0, or –l, only the least significant bit and the sign bit would be used. The remaining 14 bits would be garbage that would reduce the efficiency of the genetic algorithm. Any mutations or crossovers, these actions are described below in the section CEvolveDialog, within these bits would be meaningless and have no effect on the array output.

When the genomes are computed, either during evolution, sweep, or background stages, the document access the genome and reads the array elements to fill the magnetic field maps and resistance matrix associated with the genome. The field maps for an

element are filled according to

$$\tilde{B}_n(x,z) = \sum_{i=0}^{N-1} \xi_i B_i(x,z) \qquad [4.10]$$

where

$$\xi_i = \begin{cases} 0 & : \quad gene_i = 00,11 \\ 1 & : \quad gene_i = 01 \\ -1 & : \quad gene_i = 10 \end{cases} . \qquad [4.11]$$

The resistance matrix for the entire matrix is filled using

$$\tilde{\mathbf{R}}_{mn} = \sum_{i=1}^{N_c} \sum_{j=1}^{N_c} \xi_i \xi_j \mathbf{R}_{im} \mathbf{R}_{nj} . \qquad [4.12]$$

The field maps are stored in temporary variable arrays within the document due to their large size. Each time a new genome is selected, the field variable arrays are refilled. Since this is a simple summation, this takes little time. The resistance matrix is small enough to be stored within the genome object. If any settings are changed that would affect the SNR maps or g-factor maps, these maps are reevaluated after the field maps are filled.

CConstraintsDialog

The evaluation of an array is dependent on many parameters. The values of the parameters are edited in the CConstraintsDialog object. The dialog, shown in Figure 23, is divided into three sections; "Array Constraints", "Image Constraints," and "Sample Properties." The "Array Constraints" section controls dimensions of the basis array. Any changes made in this section will invalidate the mutual resistance matrix and the base magnetic fields and will require these values to be recalculated. This may require a considerable amount of time.

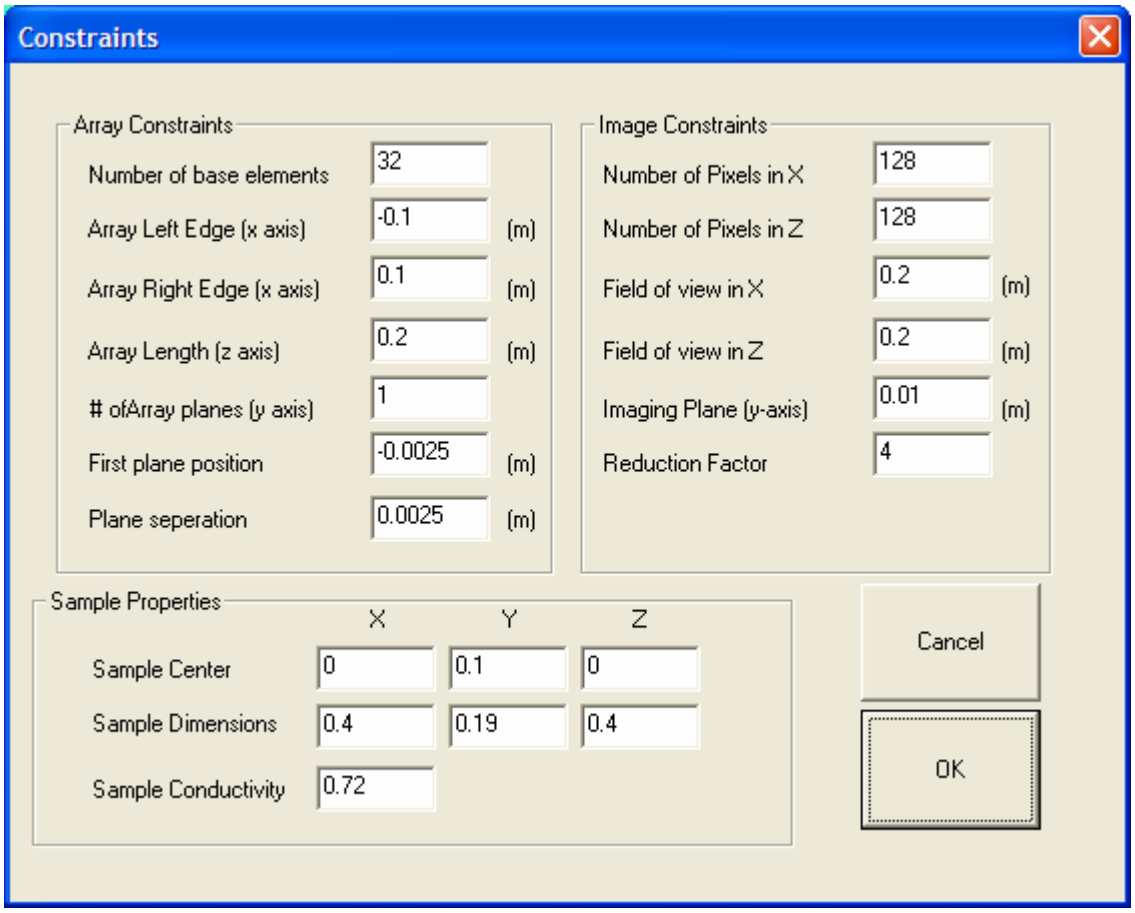


Figure 23. Dialog box for changing MV_SENSE parameters.

The "Image Constraints" section determines the plane in which the basis element magnetic field sensitivities are computed and the size of the sampling grid. Changing any parameter in this section, other than reduction factor, will invalidate the base magnetic field sensitivities and will force them to be recomputed.

The "Sample Properties" section controls the position, dimensions, and conductivity of the sample volume. It is assumed that the sample covers a rectangular

volume. Changing any parameters in this section invalidates the mutual resistance matrix for the base array and forces it to be recalculated. It is a good idea that once a new resistance matrix has been calculated for a set of parameters, that it be saved so that it can be used later.

CSweepDialog

In order to examine how an array's performance varies as imaging parameters are changed, it is useful to be able to sweep a parameter over a certain range. The CSweepDialog object, shown in Figure 24, provides this functionality. Currently, the dialog allows four imaging variables, Depth, FOV_X, FOV_Z, and Reduction Factor, to be swept. The range over which the variables are changed is set using the Start and Stop edit boxes while the number of sampling points is controlled with steps. The array genomes to be evaluated at each step are controlled using the "Genomes of Interest" edit box. The genomes can be listed numerically and separated by commas or all the genomes can be evaluated at each step by using the keyword "all". Once each evaluation step is completed, the statistical information for each genome is stored as a tab delimited text file that can be imported into either Matlab® or ExCel® for display. The evaluation is implemented using a background worker thread so that the application doesn't appear "frozen." The progress of the thread is monitored during each step, upper progress bar, and overall, lower progress bar.

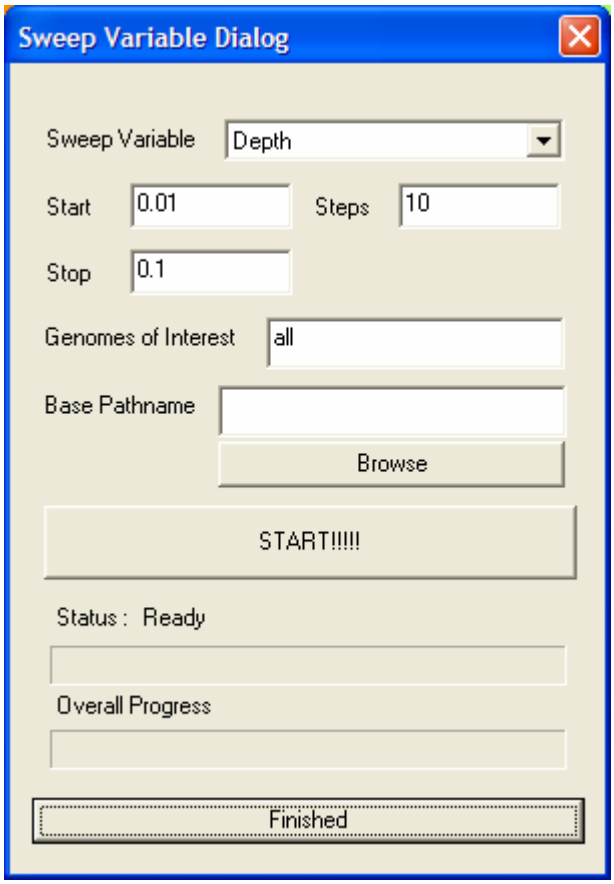


Figure 24. CSweep dialog box.

CEvolveDialog

The genetic optimization algorithm is implemented using this dialog object, shown in Figure 25. A genetic optimization was implemented due to the large solution space of the problem. A basis array with thirty-two elements would have 32^3 or 32768 possible configuration for each array element. The algorithm allows for two selection parameters as well as random mutation to an array element. The selection of which criteria to select with is controlled using two list boxes. The available criteria are the statistical information that is computed for each genome. Finally, the rate of mutation and the number of generation to propagate over are set.

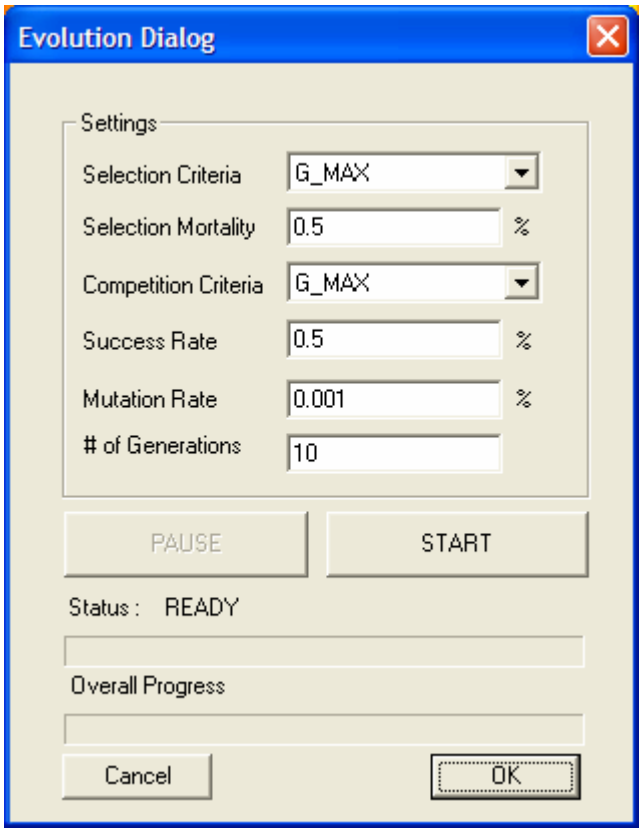


Figure 25. CEvolve dialog box.

Pressing the start button begins the evolution process of the genetic algorithm. The block diagram of this algorithm is shown in Figure 26. The algorithm implemented allows for selective pressure to be applied using two different parameters. The first parameter controls mortality of the genomes. The genomes are sorted according to the specified varied and then only a percentage is allowed to survive. The survivors are then sorted according to the second selection criteria and only the best fraction, according to this parameter, are allowed to mate. The children and the survivors are then grouped to form a new generation and the process is repeated until the number of requested generations is completed.

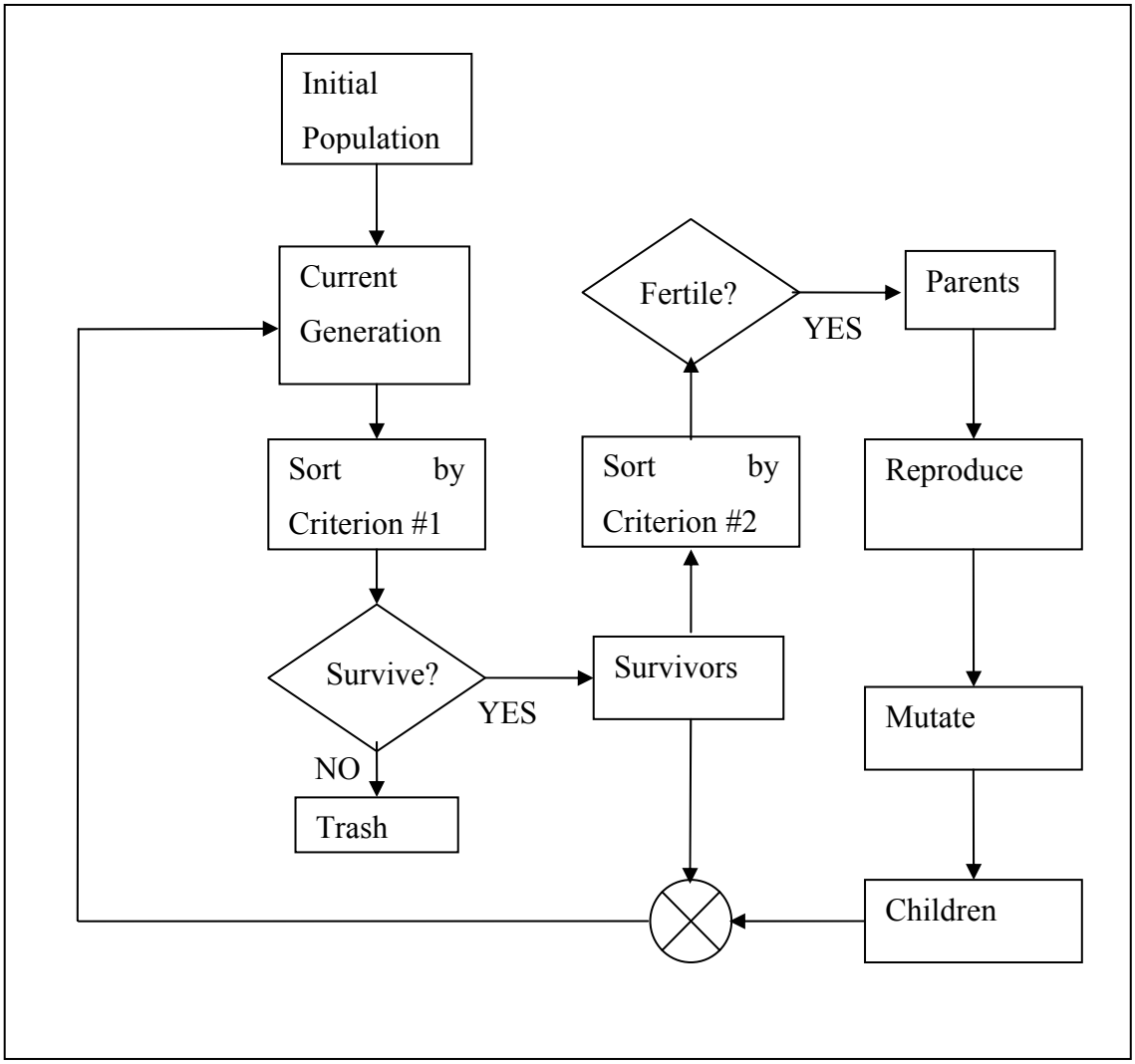


Figure 26. Genetic algorithm block diagram.

The reproduction method used in the algorithm is a double crossover. After choosing two parents to combine, two crossover points are selected for each gene. The genes are cut at these points and the information between these points is traded. A diagram of this process is shown in Figure 27. Mutation is done by randomly changing a bit on a random gene. The process of reproduction allows for the optimization to search the space already spanned by the possible solutions, while the mutation process allows the algorithm the ability to search in new directions.

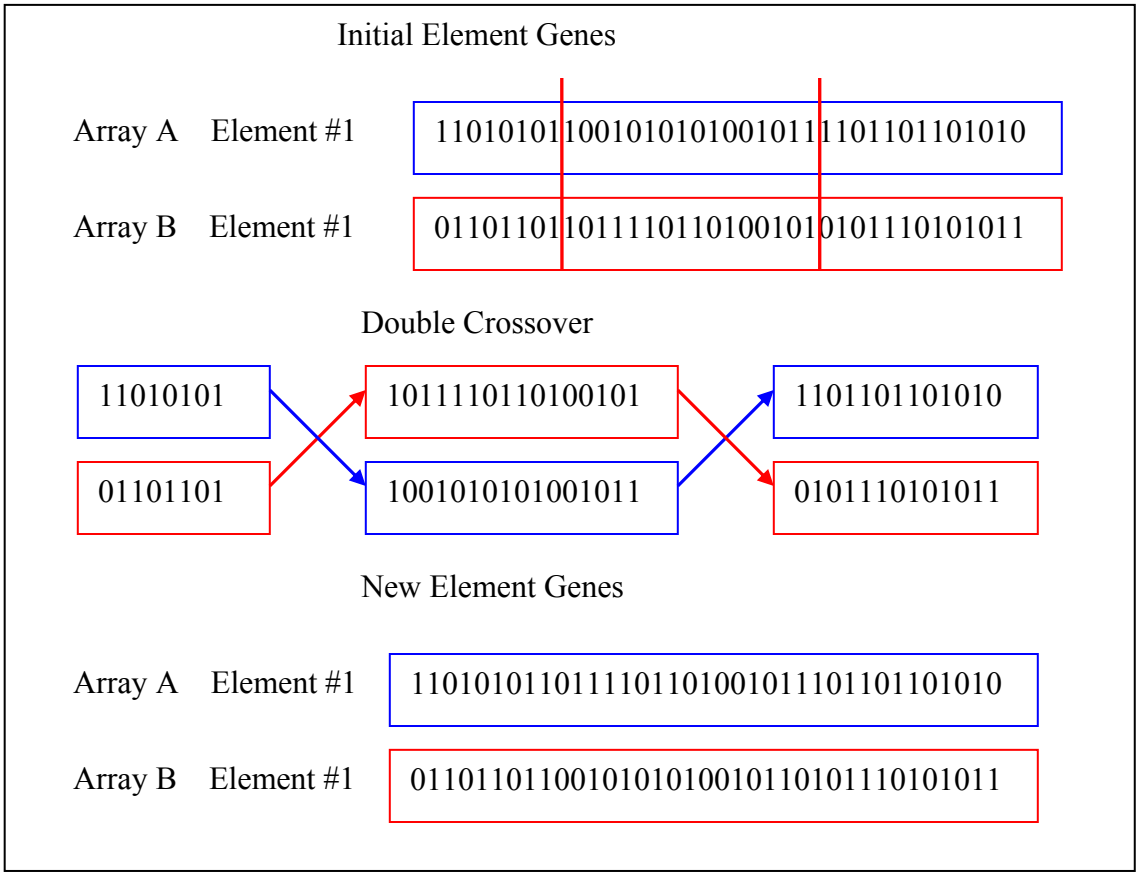


Figure 27. Double crossover for creation of new coil arrays.

Matlab

Matlab® was used to process information that required predominantly matrix calculations and to display the large amounts of information generated by MV_SENSE. Scripts were written to simulate SNR maps, g-factor maps, determine Noise correlation, optimize weighting factors and to process acquired images. Matlab® also includes functions to keep track of how many operations a program requires to complete.

MV_SENSE Interface

This script allows Matlab® to read in and process the text files generated by the Sweep Dialog box. Depending on how many arrays the dialog box was set to track, Matlab® may have to open and process up to one hundred different files at a time. The

file format has nine columns that are tab delimited. The first column lists the values of the swept variable. Columns 2-9, in order, are maximum SNR, minimum SNR, mean SNR, standard deviation of SNR, maximum g-factor, minimum g-factor, mean g-factor, and standard deviation of g-factor.

Once the files have been imported, Matlab® is used to graphically display any of the statistical variables versus the sweep variable for a selected coil array. The code can also display aggregate sums and means of variables so as to determine general trends.

SNR Maps

Scripts were written to simulate planar grid arrays as well as array of arbitrary coil geometry. The magnetic field sensitivities were computed according to Eq.[2.6] where

$$\mathbf{B}(\mathbf{x}) = \frac{\mu_0}{4\pi} \oint \frac{Id\mathbf{x}' \times (\mathbf{x} - \mathbf{x}')}{|\mathbf{x} - \mathbf{x}'|^3}. \qquad [4.13]$$

The mutual resistance matrix was generated using the C++ code and imported since Matlab® isn't very efficient at evaluating 3D integrals. Methods were written to evaluate SNR maps for a fully utilized array and for combination into a virtual array.

Full Planar Maps

The script for computing an SNR map for a planar grid array has the function header:

function

[map,grid,center]=full_snr_map(FOV,Plane,Pix,offset,Aperature,theta,coils,array_offset, R).

The function requires nine arguments and returns up to three matrices. The arguments and their definitions are listed in Table 5. The first four arguments are used to describe the imaging plane. The points contained in the imaging plane are returned in grid using

$$\mathbf{grid}_{i,j} = \mathbf{x}_0 + i\hat{\mathbf{n}}_{ro}\Delta_{ro} + j\hat{\mathbf{n}}_{pe}\Delta_{pe} \qquad [4.14]$$

where $\mathbf{x}_0$ is the point associated with the lower left corner of the image, $\hat{\mathbf{n}}_{ro}$ and $\hat{\mathbf{n}}_{pe}$ are

unit vectors describing the readout and phase encode direction, and Δ_{ro} and Δ_{pe} are the resolution in the readout and phase encode directions. The grid is returned in order to facilitate making surface plots of the SNR map.

Table 5. Argument descriptions for FULL_SNR_Map script

Variable Name	Description
INPUTS	
FOV	2 element vector containing length and width of FOV
Plane	2-by-3 matrix whose rows are unit vectors describing readout and phase encode directions
Pix	2 element vector specifying the number of pixels in the image
offset	a vector specifying the center of the image
Aperature	2 element vector describing the overall length and width of the planar array
theta	the angle the planar array makes with the x-axis
coils	2 element vector containing the number of rows and columns in the planar array
array_offset	a vector specifying the center of the planar array
R	imported mutual resistance matrix for the array
OUTPUTS	
map	2D matrix containing the SNR map
grid	3D matrix containing the image points
center	3D matrix containing the centers of the planar array elements

The next four arguments are used to construct the planar array. The centers of the planar array elements are defined as

$$\mathbf{center}_{m,n} = \mathbf{c}_0 + m\begin{pmatrix}\cos\theta\\ \sin\theta\\ 0\end{pmatrix}w + n\begin{pmatrix}0\\ 0\\ 1\end{pmatrix}l \qquad [4.15]$$

where $\mathbf{c}_0$ is the center of the lower left array element, w is the width of an array element, and l is the length of an element. The theta term is included to allow for the array to conform to a curved surface, such as a cylindrical or elliptical magnet bore. The final input argument, R, is the imported mutual resistance matrix for the array.

The SNR map is finally computed according to

$$map_{m,n} = \sqrt{\mathbf{S}_{m,n}^{\dagger}\mathbf{R}^{-1}\mathbf{S}_{m,n}} \qquad [4.16]$$

where **S** is a column vector containing all the element sensitivities at point $\mathbf{grid}_{m,n}$.

Combined Planar SNR Maps

When a large array is combined and used on a limited number of receiver channel, as discussed in SECTION, this combination must be included in calculating the SNR maps. The function declaration for this script,

function[map,grid,center]=combo_snr_map(FOV,Plane,Pix,offset,Aperature,theta,coils,array_offset,R,W)

is nearly identical, Table 6, to the full SNR map, but now has an extra input argument, W. The parameter, W, is a rectangular matrix that reduces N_c coils to N_r receiver channels. The function returns the variables grid and center as defined in [4.14] and [4.15]. The SNR map is now computed as

$$map_{m,n} = \sqrt{\mathbf{S}_{m,n}^{\dagger}\mathbf{W}^{\dagger}\left(\mathbf{W}\mathbf{R}\mathbf{W}^{\dagger}\right)^{-1}\mathbf{W}\mathbf{S}_{m,n}} \qquad [4.17]$$

Table 6. Arguments for combined_snr_map script

Variable Name	Description
INPUTS	
FOV	2 element vector containing length and width of FOV
Plane	2-by-3 matrix whose rows are unit vectors describing readout and phase encode directions
Pix	2 element vector specifying the number of pixels in the image
offset	a vector specifying the center of the image
Aperature	2 element vector describing the overall length and width of the planar array
theta	the angle the planar array makes with the x-axis
coils	2 element vector containing the number of rows and columns in the planar array
array_offset	a vector specifying the center of the planar array
R	imported mutual resistance matrix for the array
W	N_r by N_c matrix containing weighting coefficients to combine N_c coils to N_r receivers.
OUTPUTS	
map	2D matrix containing the combined SNR map
grid	3D matrix containing the image points
center	3D matrix containing the centers of the planar array elements

Arbitrary Array SNR maps

In order to evaluate the SNR map of an array of arbitrary elements, the number of array elements and the geometry of each must be defined. This is done using a geometry file that contains the vertices, the point where the current changes direction, of the coil element. For example, a square loop has four vertices. It is assumed that the current flows along a straight wire between these points. To approximate a curved line,

more vertices are used. Each coil element has its own geometry file.

Once imported, the geometry files are used to calculate the magnetic field sensitivities in a plane. The resistance matrix for the array is approximated by evaluating the magnetic vector potential over a three dimensional grid and then simply summing all the values. This is only a first order approximation and is not necessarily accurate. Using the sensitivity maps for the array, the SNR map is computed using [4.16] when a full receiver is available, or using [4.17] when the elements are combined.

G Maps

The computation of g-factor maps for planar arrays is done using the functions

function[map,grid,center]=full_g_map(FOV,Plane,Pix,offset,Aperature,theta,coils,array_offset, R, AF)

and

function[map,grid,center]=full_snr_map(FOV,Plane,Pix,offset,Aperature,theta,coils,array_offset,R,W, AF).

The input and output arguments and their definitions are given in Table 7. The input arguments are identical to the SNR map functions, except that the SENSE acceleration factor, AF, is added. The functions return grid and center, as were defined in [4.14] and [4.15], and the g-factor map. For the full receiver case, the g-factor map is computed according to

$$map_{m,n} = \sqrt{\left(\mathbf{S}^{\dagger}\mathbf{R}^{-1}\mathbf{S}\right)^{-1}_{(m,n),(m,n)} \cdot \left(\mathbf{S}^{\dagger}\mathbf{R}^{-1}\mathbf{S}\right)_{(m,n),(m,n)}} \qquad [4.18]$$

where **S** is now a N_c by AF matrix containing the coil sensitivities at the aliased points. For a combined coil array,

$$map_{m,n} = \sqrt{\left(\mathbf{S}^{\dagger}\mathbf{W}^{\dagger}\left(\mathbf{W}\mathbf{R}\mathbf{W}^{\dagger}\right)^{-1}\mathbf{W}\mathbf{S}\right)^{-1}_{(m,n),(m,n)} \cdot \left(\mathbf{S}^{\dagger}\mathbf{W}^{\dagger}\left(\mathbf{W}\mathbf{R}\mathbf{W}^{\dagger}\right)^{-1}\mathbf{W}\mathbf{S}\right)_{(m,n),(m,n)}} \; . \qquad [4.19]$$

Table 7. Argument definitions for planar g-map functions

Variable Name	Description
INPUTS	
FOV	2 element vector containing length and width of FOV
Plane	2-by-3 matrix whose rows are unit vectors describing readout and phase encode directions
Pix	2 element vector specifying the number of pixels in the image
offset	a vector specifying the center of the image
Aperature	2 element vector describing the overall length and width of the planar array
theta	the angle the planar array makes with the x-axis
coils	2 element vector containing the number of rows and columns in the planar array
array_offset	a vector specifying the center of the planar array
R	imported mutual resistance matrix for the array
W	N_r by N_c matrix containing weighting coefficients to combine N_c coils to N_r receivers.
AF	SENSE acceleration, or reduction, factor
OUTPUTS	
map	2D matrix containing the combined SNR map
grid	3D matrix containing the image points
center	3D matrix containing the centers of the planar array elements

For arbitrary arrays, the coil geometry files are read and fields computed exactly as in the arbitrary SNR case. Then the g-factor maps are computed using [4.18] or [4.19] depending if the coils are used independently, or combined.

Imaging

Matlab® was used to process imaging data acquired on the MR scanner. The data was imported using a script, *omegaread*, that reads the omega data files. The images are then processed using a 2D fast fourier transform, or a SENSE reconstruction, and then displayed.

CHAPTER V

RESULTS

The results section is broken into several sections. First baseline results from a single loop are calculated for comparisons. Then, field maps and isosurfaces are evaluated for large arrays with full sets of receivers and compared with the single loop to show relative gain in performance in order to provide a reference. In the next section, results from "constructed" arrays are shown and compared with a full receiver using all the basis functions. Next the effects on the SNR of combining a large array into a limited number of channels are shown. Finally rapid image results are shown that show different optimization points, different phase encode direction in an attempt to establish trends and make generalizations.

Validation

The electromagnetic models used calculate the fields of a 64 element linear array of planar pairs(21) and these calculations were compared against measured data. Using the MRI scanner, an image was acquired 1mm above the plane of the array. By taking a profile through a homogenous region of the object being image, the magnitude and phase of a coil element in the array was measured. Since the region is homogenous, the acquired profile represents the sensitivity of the coil element. In Figure 28, the measured sensitivity is compared with the computed sensitivity for the coil element. The graphs show close agreement in both magnitude and phase of the sensitivity profile.

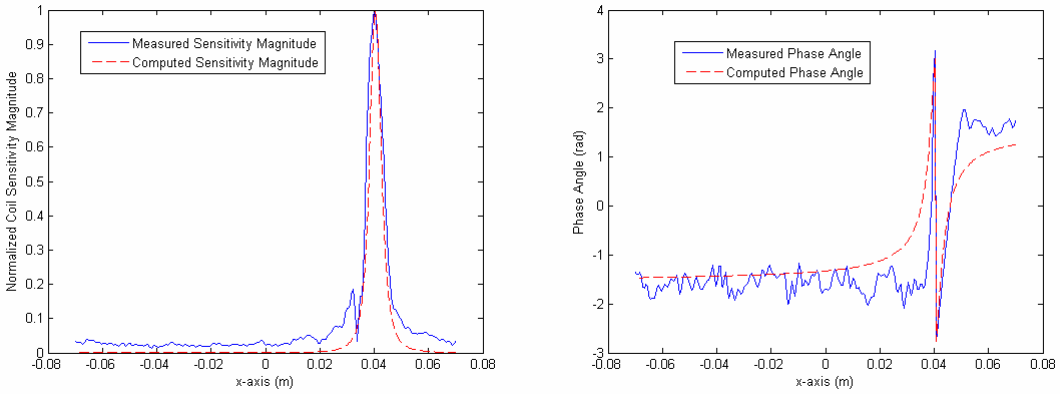


Figure 28. Measured versus computed validation of electromagnetic model.

In order to validate the algorithms for computing SNR and mutual resistance, a plot is recreated from Wright's paper on array combination (47) for spectroscopy. The plot is shown in Figure 29 and shows how the optimal SNR falls off with distance from the array plane for equally size arrays with different numbers of elements. In this case, the overall array size was 12.375cm by 12.375cm. The region was then subdivided into a grid where each square in the grid is a coil element. The SNR for the grid array is computed and then normalized by the SNR computed for a single loop the same dimension as the array. Close to the plane of the array, the 8x8 grid array has nearly 30 times the SNR of the single loop, while the 2x2 grid array is approximately 5 times the SNR of the loop. As the distance from the array is increased, the SNR for the different configurations converge to be roughly equal.

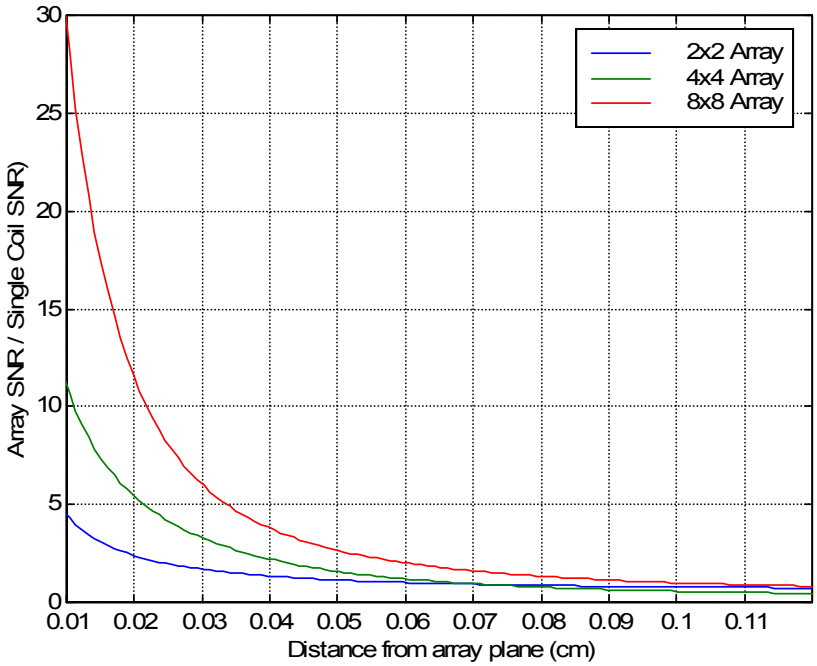


Figure 29. Comparison of array SNR to coil SNR perpendicular to array surface.

Simple Array Combination

Using the program MV_SENSE discussed earlier, arrays were constructed assuming that the current on each element was equal to one, zero, or minus one. This would ensure that current were conserved and the combined coil element could be constructed with a single length of wire. This would allow the arrays to then be constructed on a printed circuit board if required and then used for real.

This section focuses on four types of linear arrays, the global array, a local array composed of loops, a local array composed of planar pairs, and stochastic arrays. A global array is composed of elements that have sensitivity over the entire image field of view. Local arrays are comprised of elements that have sensitivity over a localized region of the field of view. A stochastic array is composed of randomly constructed elements. It is formed by randomly adding or subtracting elements from the larger base array. An individual element in the stochastic array may have a global or local

perspective.

In this section, each of the arrays is described and then comparisons between the arrays are made.

Global Array

The global array topologies are constructed so that each elements' FOV encompasses the field of view. The genome of an 8 element global array is shown in Figure 30. This array is constructed by combining the elements of a 32 element linear array. Blue sections represent current in clockwise direction while red sections are counter-clockwise. The coils range from a large loop utilizing the entire array surface, highlighted in yellow, which has deep penetration into the sample to an element that has eight twists and is sensitive close to the plane of the array.

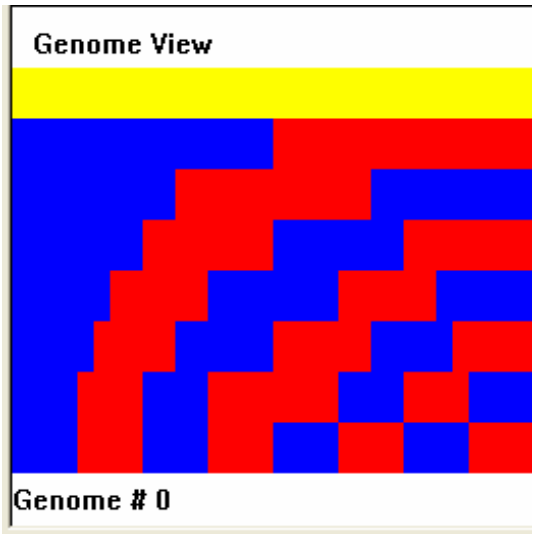


Figure 30. Genome view of an eight element global array.

The field patterns for each of these elements are shown in Figure 31 for a distance of 1cm above the array surface in order to show their variation. The regions of highest sensitivity, red, occur where the wire crosses over and the current reverse its

direction. This results in a double of the current, and its sensitivity, along this line. The large loop, coil 1, has its greatest sensitivity at the edges of the FOV and is relatively weak in the center. As twists are added to the structure in other elements, the uniformity of the sensitivity patterns improves over the imaging plane.

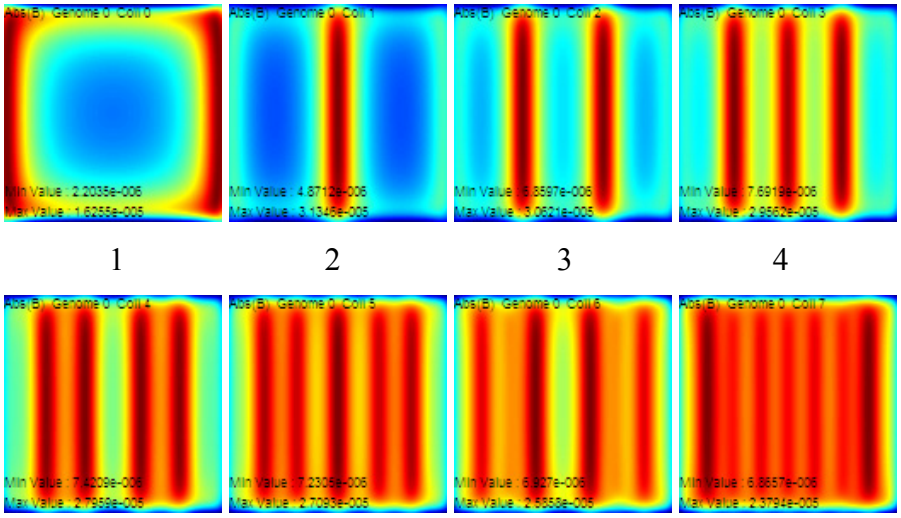


Figure 31. Element patterns for global array 1cm above array surface.

Combining the elements together for optimal SNR at an imaging depth of 1cm results in the in the SNR map shown in Figure 32. Within the center of the FOV, the SNR map is relatively uniform with hot spots near the edges. Outside the array, the SNR begins to drop off rapidly.

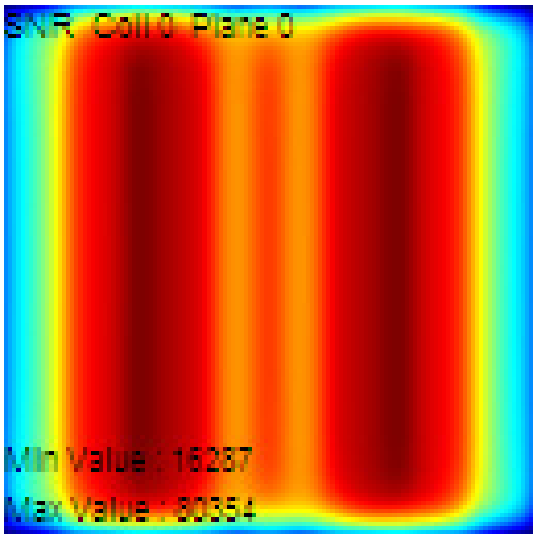


Figure 32. Combined SNR for 8 element global array at 1cm depth.

When used for SENSE imaging, this array performs reasonably well in this imaging plane. G-factor maps showing the amount of noise amplification that occurs in the image are shown in Figure 33 for reductions factors of two, left, and four, right. At the lower reduction factor, the noise is increased by about 2% at most. At a reduction factor four, the noise in the image is increase by nearly 50%. The noise amplification occurs along vertical lines do to the symmetry of the array.

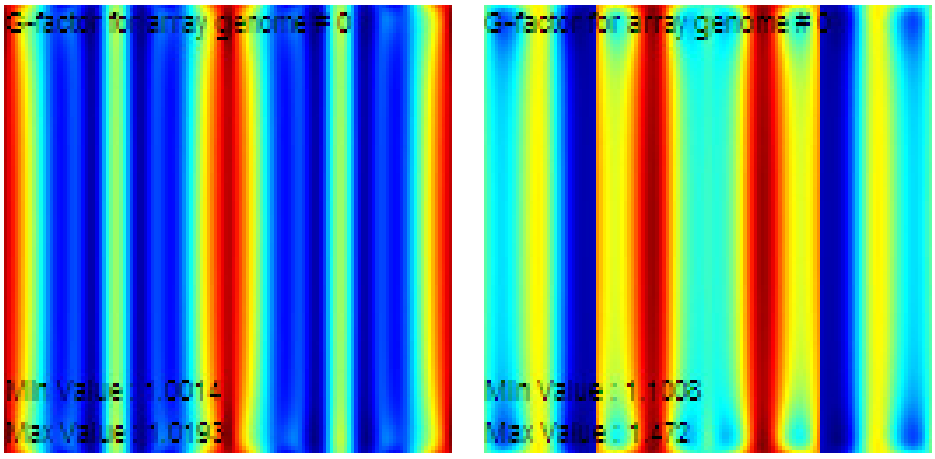


Figure 33. G-factor arrays for global array at 1cm.

Local Array with Loops

A more common method for building linear arrays is to place simple loop elements side by side. Using MV_SENSE, the loops are formed from the 32 element base array by simply adding four adjacent elements in phase. The entire genome for an eight element linear array is shown in Figure 34. The magnetic from the eight loops is shown below in Figure 35. Obviously, each elements' sensitivity is localized to a portion of the image FOV.

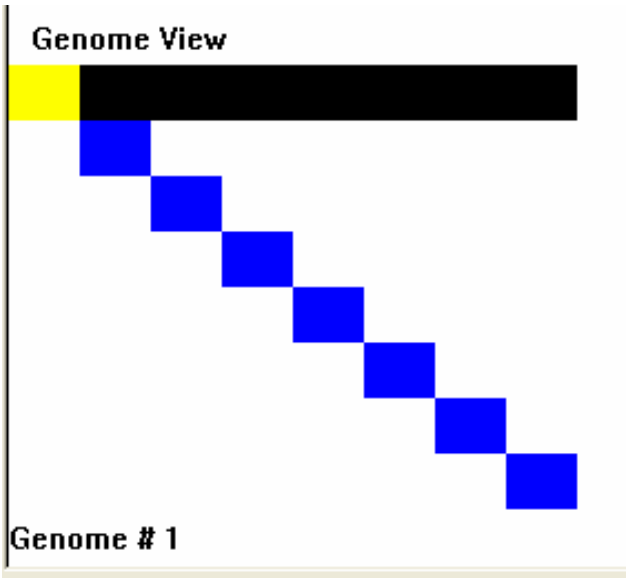


Figure 34. Genome view of eight element local loop array.

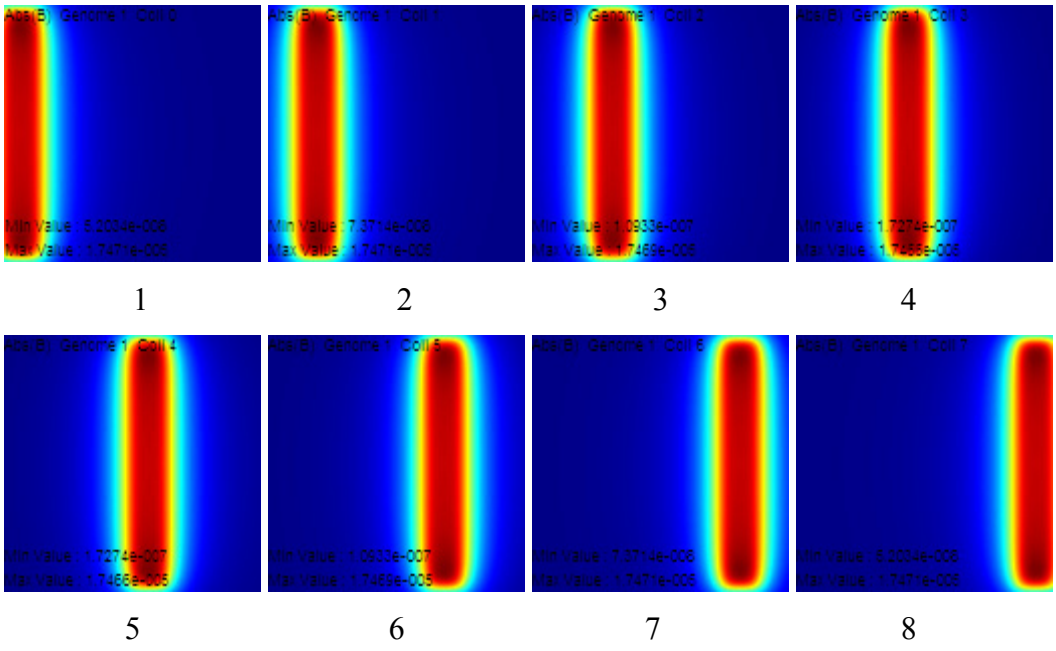


Figure 35. Field Sensitivity maps for local loop array elements.

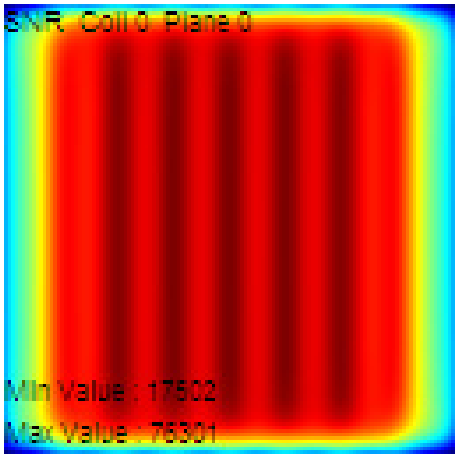
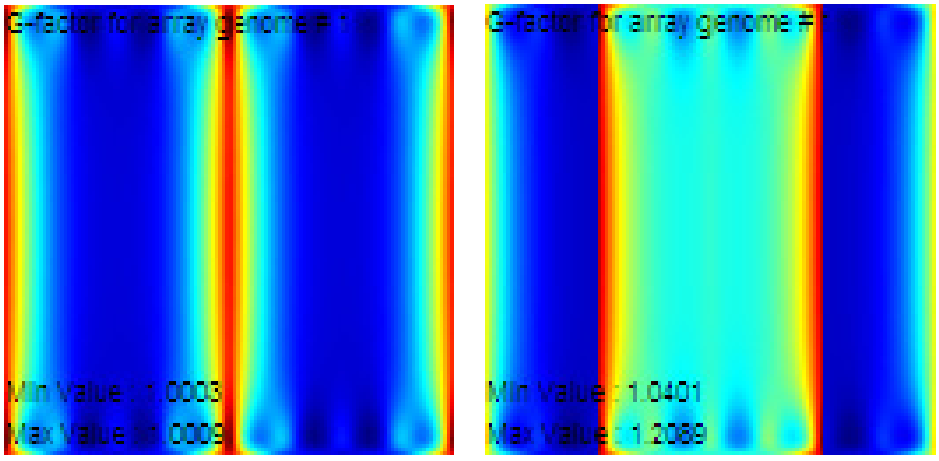


Figure 36. SNR map of combined local loop array.

Combining the local loop array to give optimal SNR over the image, Figure 36, shows that this array yields a more uniform sensitivity at this depth than the global array. However, the maximum SNR of the array at this depth is slightly less. The g-factor maps for this array, Figure 37, at a depth of 1 cm show a negligible increase in noise at a

reduction factor of two for the local loop array. At a reduction factor of four, the noise amplification due to the array is at most 20%. However, the noise increase in the local loop array is more uniform over the center of the FOV than in the global array making it more desirable.



(a) (b)

Figure 37. G-factor maps for loop array.

Local Array with Planar Pairs

A second type of local array is a linear array comprised of planar pairs instead of loops. The planar pair, or butterfly coil, is made of two adjacent loops with opposite currents. The planar pair constrains the fields of the coil element and helps to decouple adjacent elements. In MV_SENSE, the planar pair is constructed from the 32 element base array using four adjacent elements. The two elements on one side have the same phase and are added to the two elements on other side which are anti-phase. Graphically, the local planar pair array is shown in Figure 38.

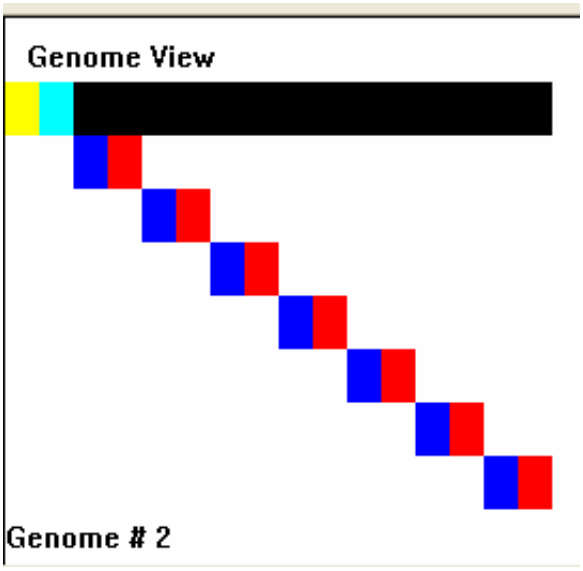


Figure 38. Genome view of local planar pair array.

The field maps for the coils elements, Figure 39, were computed at an imaging depth of 1cm. Comparing these sensitivity maps from the planar pairs to the corresponding sensitivity maps from the loops, it is seen that the planar pair's sensitivity is more confined to directly over the coil element. This greater localization reduces the coil coupling and noise correlation between array elements.

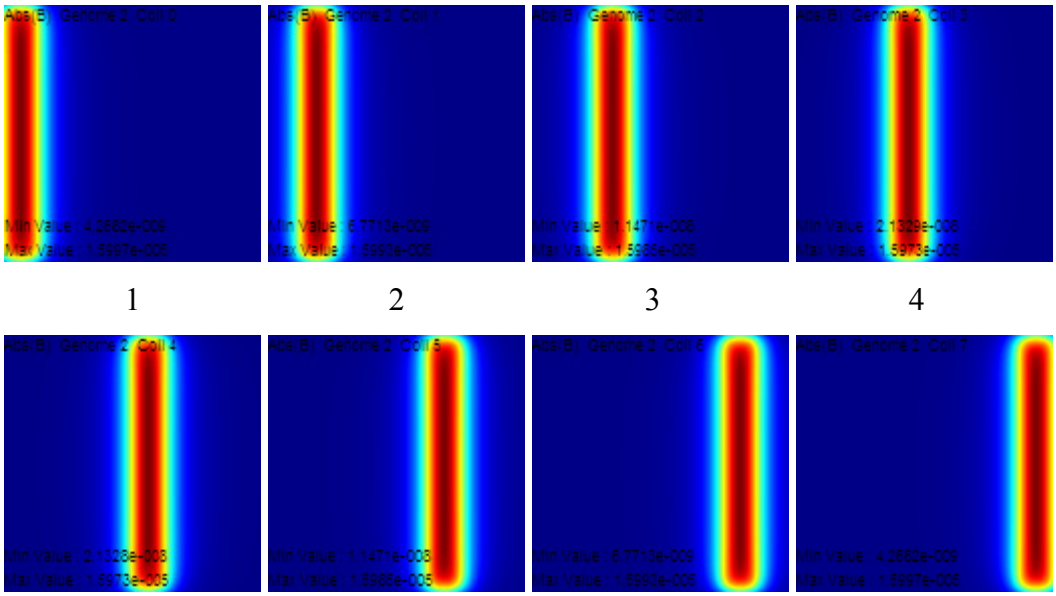


Figure 39. Field Sensitivity maps for planar pair array.

Combining the array to produce optimal SNR, Figure 40, over the imaging plane shows better homogeneity over most of the field of view than either the global array or local loop array at this depth. The maximum SNR of the array in this plane is about 10% higher than for the global array. This is attributable to the reduced noise correlation between the array elements.

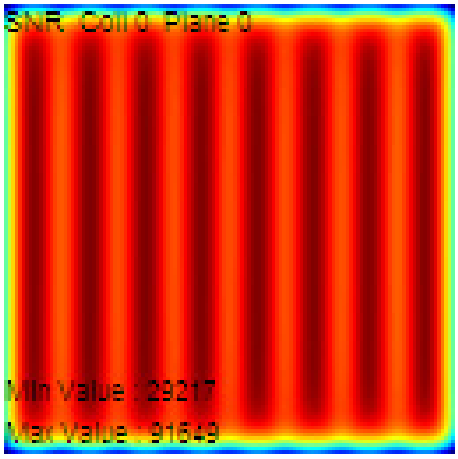
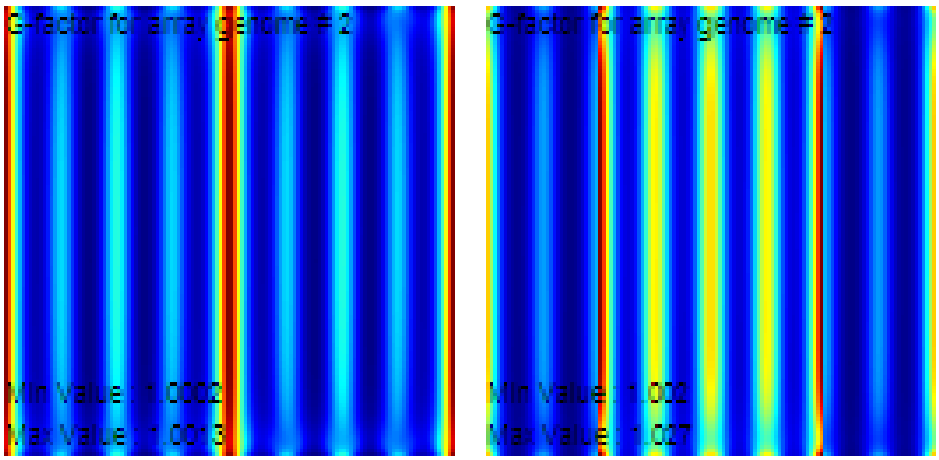


Figure 40. SNR map of combined local planar pair array.

The g-factor performance of the local planar pair array at an imaging depth of 1cm is shown in Figure 41. At a reduction factor of two, (a), the noise amplification of this array is negligible with a maximum gain of 0.13%. At a reduction factor of four, right, this array performs much better than the two previous cases with a maximum increase in noise of 2.7%.



(a) (b)

Figure 41. G-factor map of linear planar pair array.

Stochastic Array

A stochastic array is comprised of elements that are formed from random combinations of the base array. Initially, the stochastic arrays were used as seed points from which to start the genetic optimization algorithm. The algorithm would work to search through the space of all possible array combinations to find the best combinations for either SNR or g-factor. It was also thought that the lack of symmetry within the array might improve the g-factor since this may provide a better basis of support for the SENSE algorithm.

An example of an eight element stochastic array is shown in Figure 42. Each of the elements was formed randomly. The sensitivity maps for the eight coil elements are shown in Figure 43. The sensitivies of these coil elements are very disorganized.

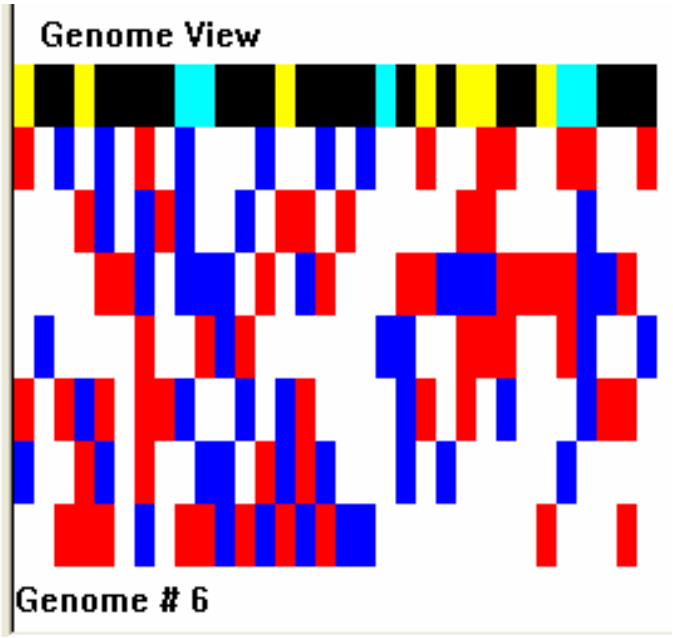


Figure 42. Genome view of stochastic array.

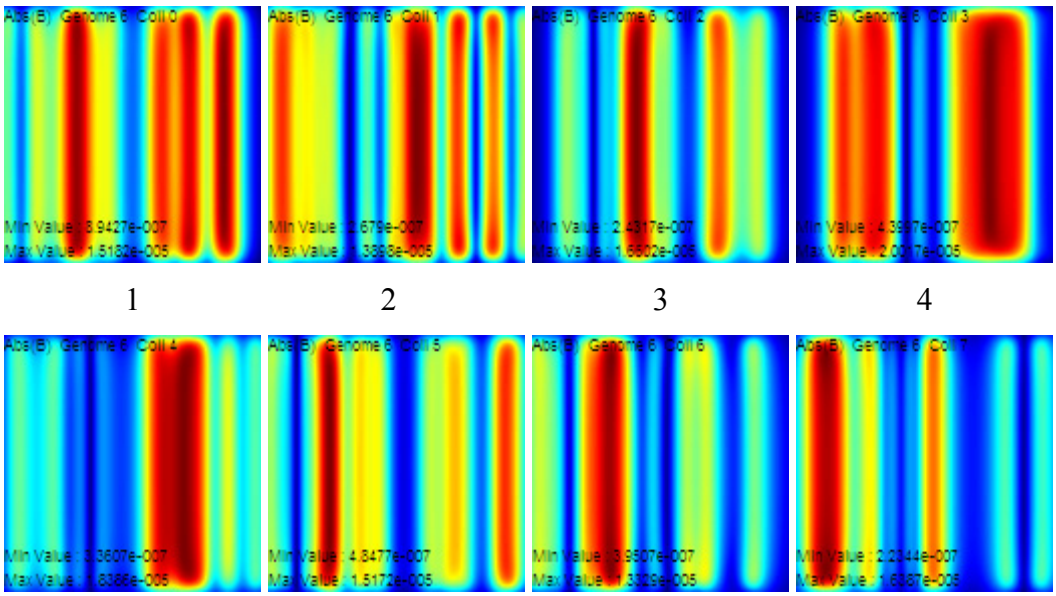


Figure 43. Sensitivity maps of a stochastic array.

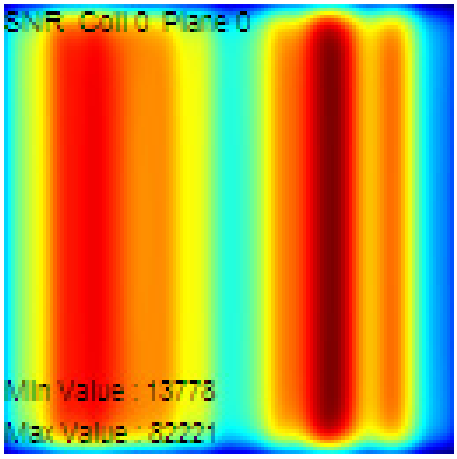


Figure 44. SNR map of combined stochastic array.

The optimally combined SNR map for the eight element stochastic array is shown in Figure 44. The maximum SNR for the array is equivalent to the maximum SNR of the global array. However, the homogeneity of the array is not very good. The sensitivity of the array is obviously focused on the right half of the image plane. This lack of uniformity is due to the irregular element sensitivities.

The g-factor maps for this stochastic array at reduction factors of two and four are shown in Figure 45. The g-factor performance of this particular stochastic array is poor at a reduction factor of two, but is comparable to the global array at a reduction factor of four.

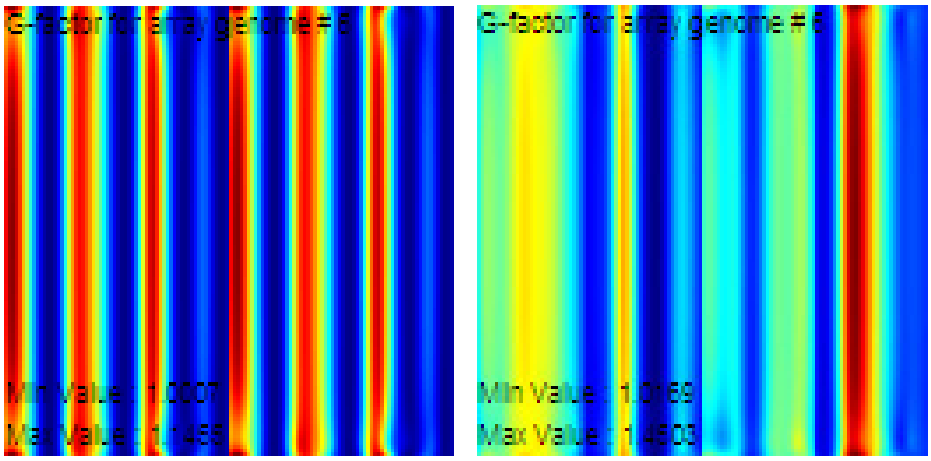


Figure 45. G-factor map of stochastic array, R=4.

Comparison of Simple Arrays

Based on the SNR maps and G-factor maps of the previous sections, it appears that these array types perform quite differently. However these maps were all computed at an imaging depth of 1cm. A more useful comparison of the arrays would be to plot their performance as imaging depth is increased. In this section, the arrays are compared against each other and against the 32 channel base array in terms of SNR performance and g-factor performance.

A log plot of maximum SNR versus imaging depth for the previously discussed array types is shown in Figure 46. For comparison, the absolute optimal for the base array is also plotted. Near the array surface, the 32 channel base array, as expected, generates the highest SNR, followed by the planar pair array as the next high in SNR. The global array, local loop, and stochastic arrays are nearly identical near the surface of the array and converge to the 32 channel optimal as the imaging depth is increased. The local planar pair array does not perform well at depth due to the inherent cancellation of the fields at depth of the planar pair. Apart from the planar pair, all the arrays performed similarly in terms of SNR performance.

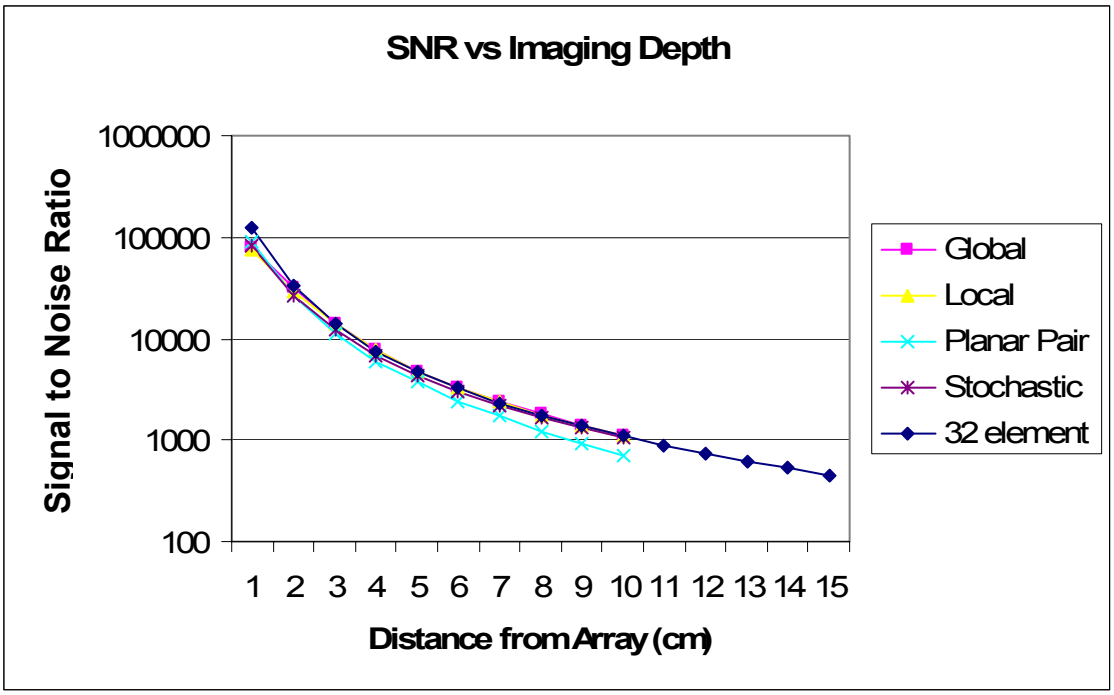


Figure 46. Comparison of simple array combination SNR versus imaging depth.

The maximum g-factor for all the arrays was computed versus imaging depth for a reduction factor of four and plotted in Figure 47. The greatest variation between the arrays occurs near the surface of the array plane. The 32 channel base array and the planar pair array perform nearly optimally with a g-factor nearly equal to one. The local loop array performs midrange among the array types while the global and stochastic arrays perform similarly close to the array. As the imaging depth is increased. The curves begin to converge. The global and local loop arrays converge rapidly, within 5cm, to be nearly equal to the 32 element optimal, while the planar pair and stochastic arrays slowly oscillate and converge toward the optimal. The slope of the linear region of the g-factor curve for this plot is 25. At 14cm, the planar pair local array actually begins to outperform the 32 channel optimal, however the SNR loss of the planar pair array at this depth is considerable and more the offsets the gain in g-factor.

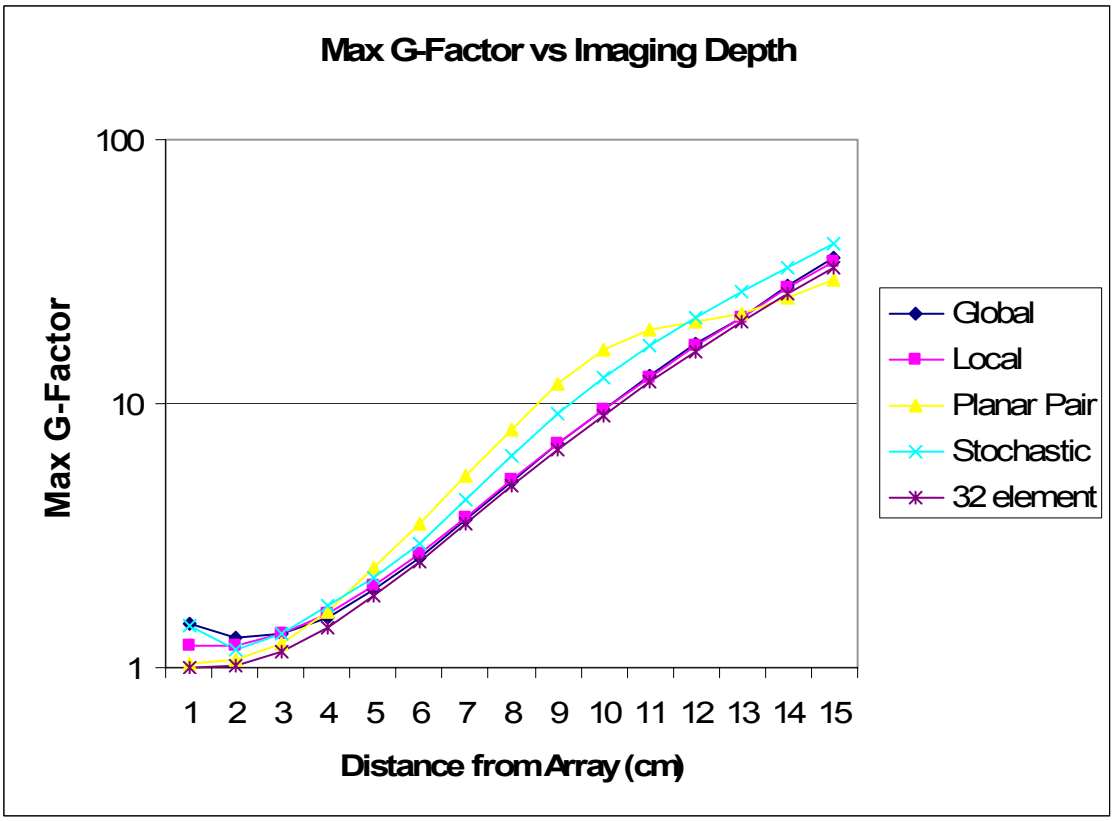


Figure 47. Maximum g-factor for simply combined arrays versus imaging depth.

In terms of minimum g-factor, the arrays again perform similarly. A log scale plot of the minimum g-factor for the arrays is shown in Figure 48. Here, the 32 element base array, the local planar pair array, and the stochastic array perform nearly perfect close to the surface of the array. The local loop array and global array show slightly higher minimum g-factors. Within 4cm of the array surface, the minima have all converged, however at 5cm the planar pair array and stochastic array begin to deviated substantially for the worse. At 11 cm, the curves have converged once again with the planar pair array performing slightly better than the 32 channel array. However this slight improvement in g-factor of the planar pair over the 32 channel array is more than offset by the losses due to the poor penetration depth of the planar pair element.

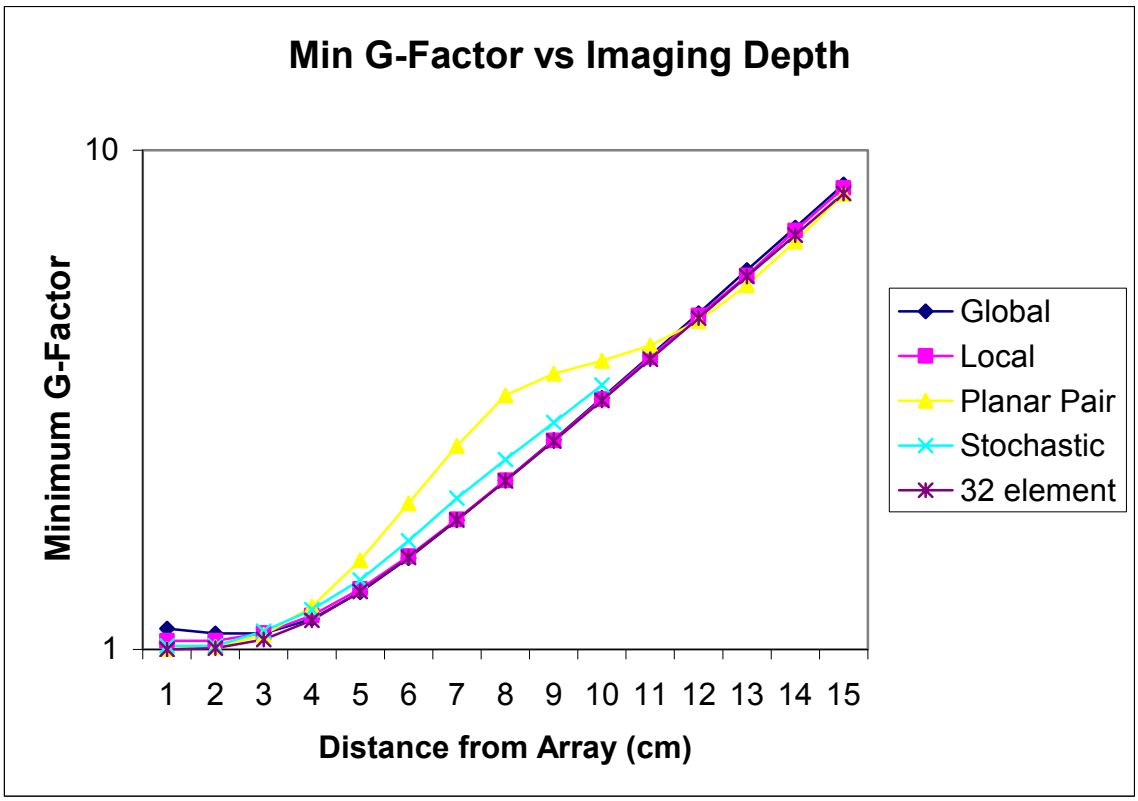


Figure 48. Minimum G-factor for simply combined arrays versus imaging depth.

Optimally Combined Array at Points

In this section, large arrays that are combined to form an array of current sheets that optimize the SNR at various points are examined. The method for determining the weighting coefficients to combine the array elements to produce the optimal SNR at a point was discussed in Chapter III. In this section, the optimal performance of the base array is discussed using a full complement of receivers is first discussed, then the SNR maps for various numbers of receivers and point combinations are computed and compared with the optimal full channel combination of the physical array. Comparisons are made in both terms of SNR and g-factor.

Signal to Noise Ratio

The base array used in these models has an overall aperture dimension of 12.375cm by 12.375cm square and lies in the coronal, x-z, plane at y=0. The array surface is subdivided into an 8x8 grid of 64 loop elements. The loops are adjacent and do not overlap. The mutual resistance matrix for the array is computed using the program RMAT.EXE discussed in the Implementation chapter. The sample volume for computing the resistance matrix was a cube, one meter on a side, positioned 5mm above the plane of the array. The conductivity of the sample volume was set to .72 S/m, approximately equal to the model given in (47) to provide comparison.

An SNR isosurface plot for the 8-by-8 planar grid array is shown in Figure 49. The region of maximum SNR is constrained, as expected, tightly to the array. Comparing this to the single loop with the same overall dimensions, Figure 50, the region within approximately 4cm of the array, enclosed by black isosurface, has over eight times the SNR of the square loop. At 10cm away, the SNR is still twice that of the single loop.

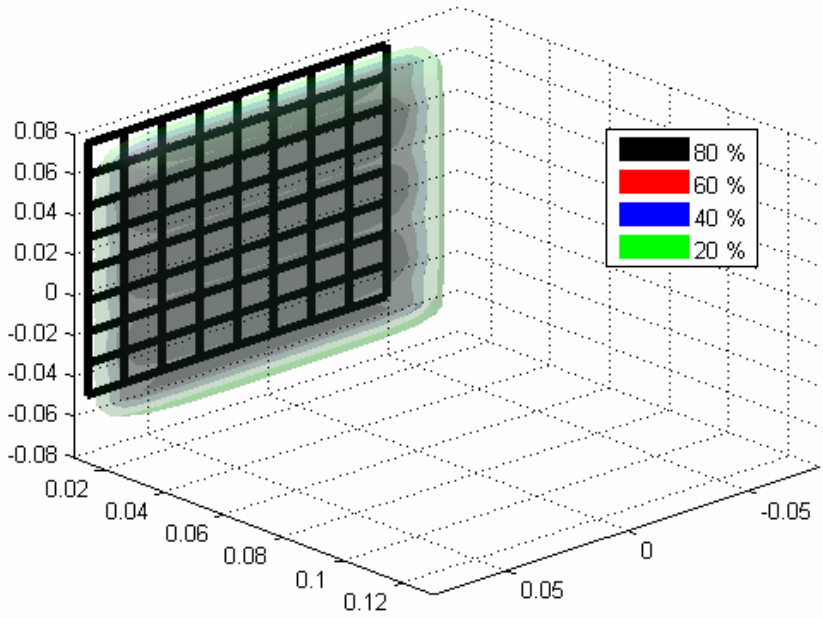


Figure 49. SNR isosurfaces for optimally combined 8-by-8 grid array.

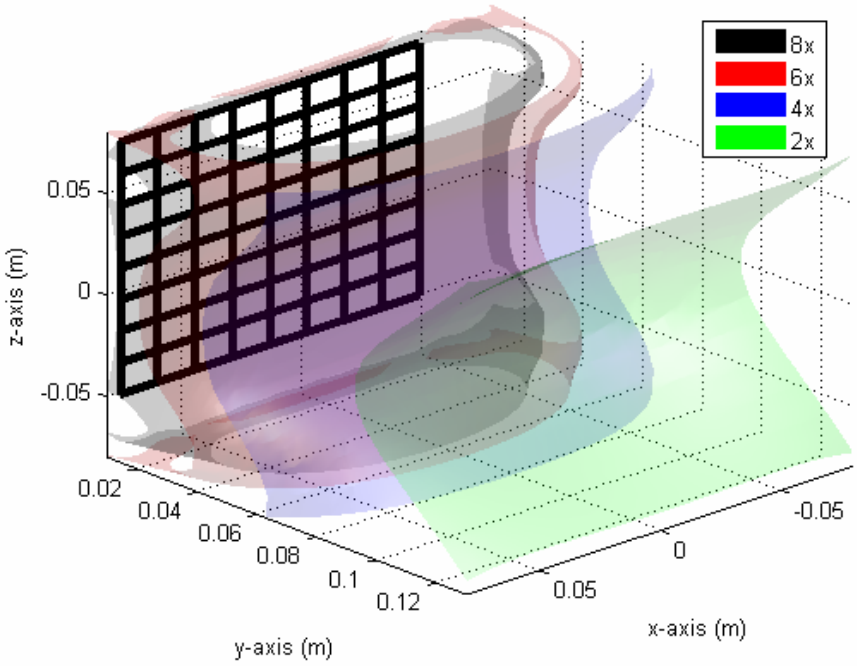


Figure 50. SNR of 8-by-8 grid array compared to single loop.

A diagram of the model is shown in Figure 51. The imaging plane lies in the coronal plane parallel to the plane of the array. The field of view of the imaging plane is 12cm x 12cm. The resolution of the image is 128x128 pixels. SNR maps and g-factor maps for the array were computed using the Matlab® functions previously discussed in Chapter IV.

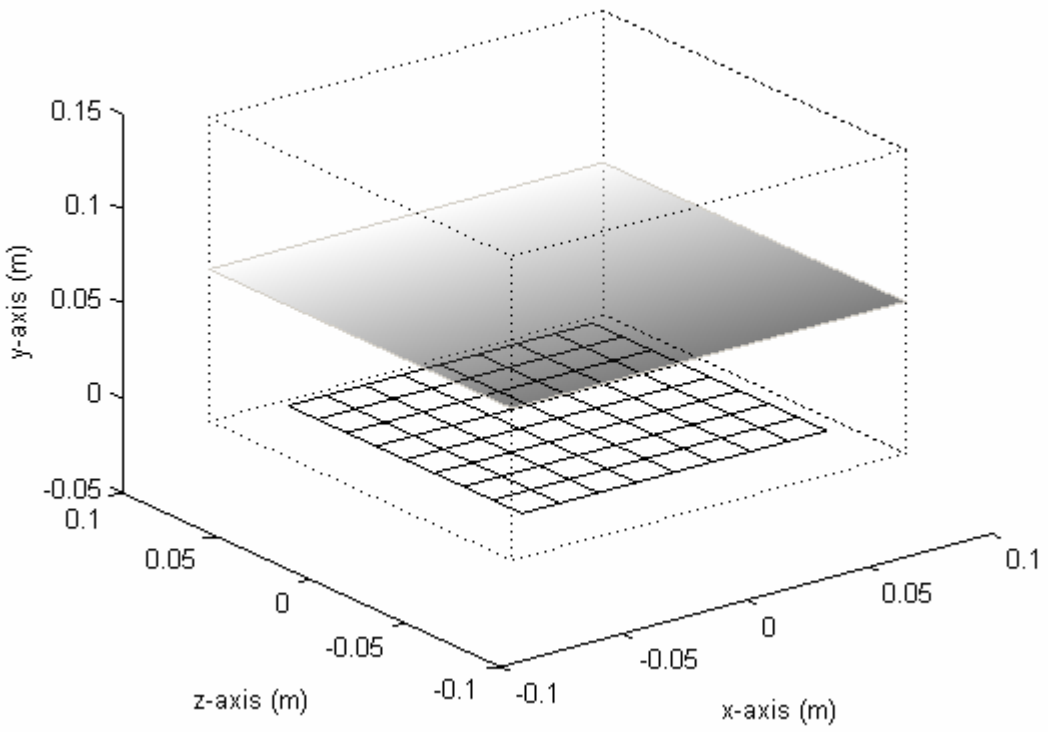


Figure 51. Diagram of geometry used.

The SNR map of the fully combined array using 64 channels at an imaging depth of 1.5cm is shown in Figure 52. The average improvement over the plane, as compared to a single loop, is a factor of 12. The greatest improvement over the loop occurs at the +/- z edges of the FOV. This is because the effective magnetic field of the loop in these regions is predominantly $\hat{\mathbf{z}}$-directed and doesn't contribute to the MR signal. However, in these regions the array the effective magnetic field of the combined array has much larger tangential component that is sensitive to the MR signal. The four asterisks denote

the points chosen to implement a combination of the array using just four receiver channels.

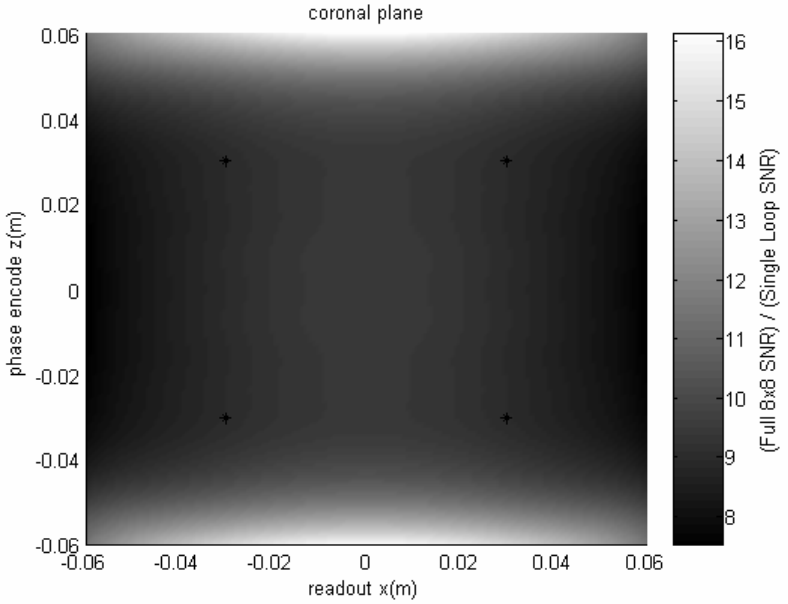


Figure 52. 64 element 8x8 grid array SNR map using 64 receivers.

The array elements were combined down to four receiver channels. The weighting coefficients for the channels were chosen to yield the optimal SNR at four equally spaced points in the image FOV. The coefficients were computed using Eq.[3.24]. The combined SNR map using 4 channels is shown in Figure 53. This SNR map has been normalized by the optimal SNR map using 64 channels, Figure 52. At the four chosen points, the normalized SNR map is equal to one meaning the SNR is optimal for the array at these points. Away from these points, the SNR falls off rapidly, particularly in the z-direction. At its lowest point, the SNR of the image using only four channels at these four points is less that 30% of the optimal SNR.

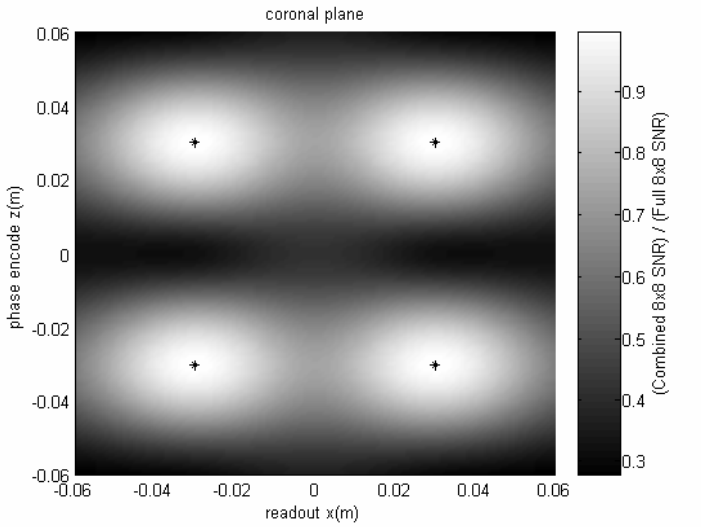


Figure 53. 64 element 8x8 grid array combined using 4 receivers.

In order to improve the coverage area of the sensitivity, four more receivers and points are added to make a total of eight receivers. The normalized SNR map of 64 elements combined down to eight receivers is shown in Figure 54. The chosen points form a square two-thirds the size of the image FOV. At the points, the SNR is equal to

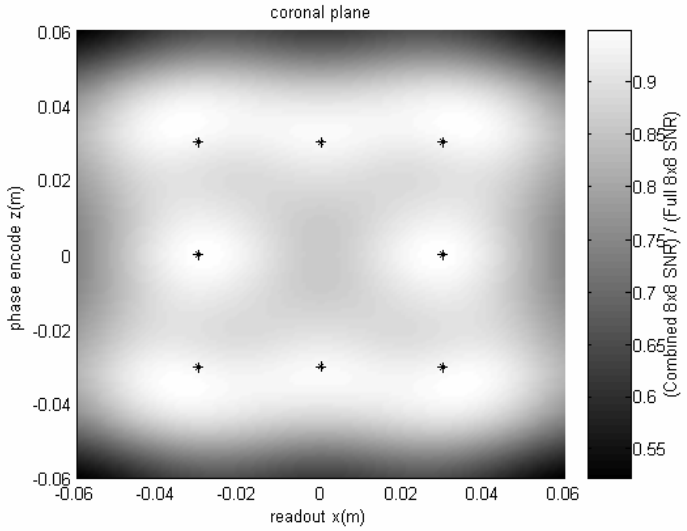


Figure 54. 64 element 8x8 grid array combined using 8 receivers.

the SNR of the optimally combined 64 element array. With the addition of the extra receivers, the coverage of the point combined array is more uniform than when just four channels were used. The minimum SNR is approximately 50% of the optimal SNR map, however the absolute minimum has been confined to the outside edges of the FOV away from the image information. In the center of the FOV, the SNR is about 75% of optimal.

Clearly, using more receivers improved the image quality. In order to quantitatively measure the differences between configurations, the root mean square deviation, $\in$, is defined as

$$\in = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(\frac{SNR\_opt_n - SNR\_cmb_n}{SNR\_opt_n}\right)^2} \qquad [5.1]$$

where N is the total number of pixels in the image, *SNR_opt* is the optimal SNR for the array, and *SNR_cmb* is the SNR of the combined array using fewer receivers. SNR maps for one, two, four, eight, sixteen, thirty-two, and sixty-four receivers are then computed using a 9x9 grid array. The dimensions and properties of the array are identical to the array discussed earlier except that it has been subdivided into a 9x9, instead of an 8x8, grid. The greater number of elements allows a greater number of receivers to be examined. For one channel, the point chosen is in the exact center of the field of view. For the other receiver configurations, the chosen points are shown in Figure 55.

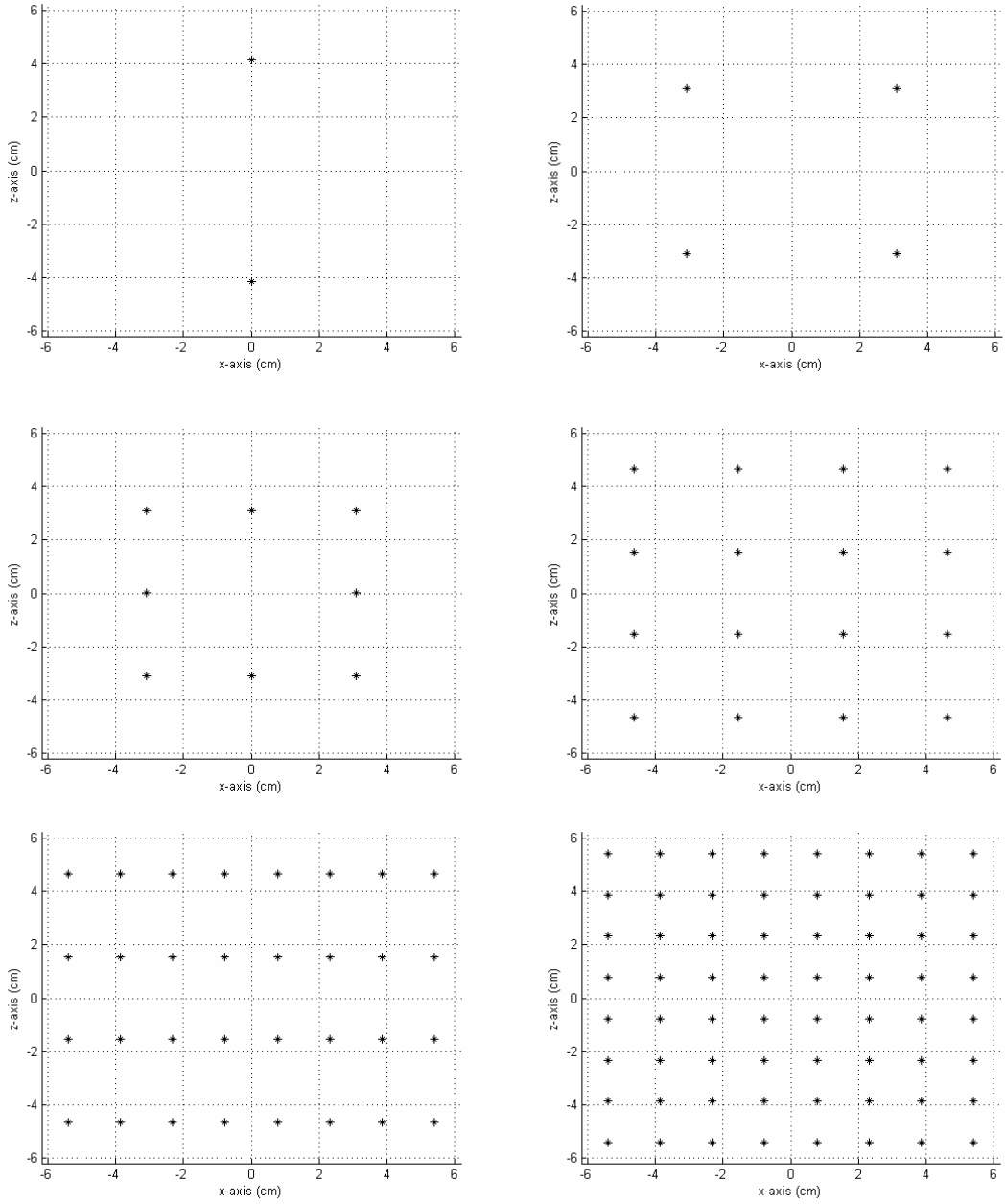


Figure 55. Points for which receivers channels are optimized.

The deviation from the optimally combined array is computed for each configuration as the imaging depth is increased. The results are shown in Figure 56. From the plot, it is seen that the more receivers present, the greater the SNR as is expected. The optimally combined array requires 81 independent receiver channels.

The plot shows that at a depth of 2cm, the 64 channel configuration performs just as well. The 32 and 16 channels reach optimal performance at 8cm, while the 8 channel converges at 13cm. The remaining configurations converge farther out, but at this distance the overall SNR of the array is becoming negligible. An interesting observation in the plot is how the traces are grouped into four groups. The groups appear to correspond with the addition of a new row of points in the $\hat{\mathbf{z}}$-direction. One channel obviously only has a point in the center of the FOV. The two, four, and a eight channel group only has two full rows in the $\hat{\mathbf{z}}$-direction, as shown in the previous figure. The sixteen and thirty-two channel arrays have four full rows in $\hat{\mathbf{z}}$ and the sixty-four channel array has eight rows. This implies that it may be better to have more rows in $\hat{\mathbf{z}}$ than columns in $\hat{\mathbf{x}}$ in order to improve SNR due to the insensitivity of the $\hat{\mathbf{z}}$ component of the magnetic field to the MR signal.

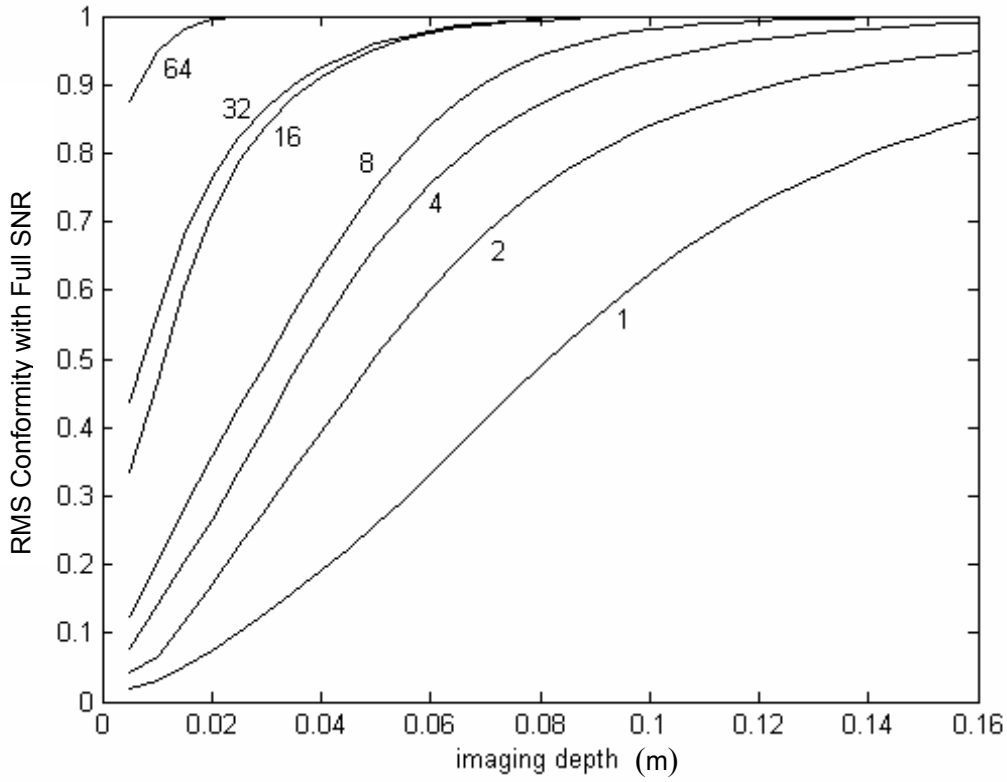


Figure 56. SNR performance of point combined 9x9 12.5cm square grid array.

In order to show the scalability of these results, the computation was repeated for a 25cm square grid array. The spacing for the chosen points for the receivers' channels are doubled as well. The plot for this double sized array is shown in Figure 57. The plots are similar with slight differences in the paths of the traces showing that the results are extendable to arrays of different sizes.

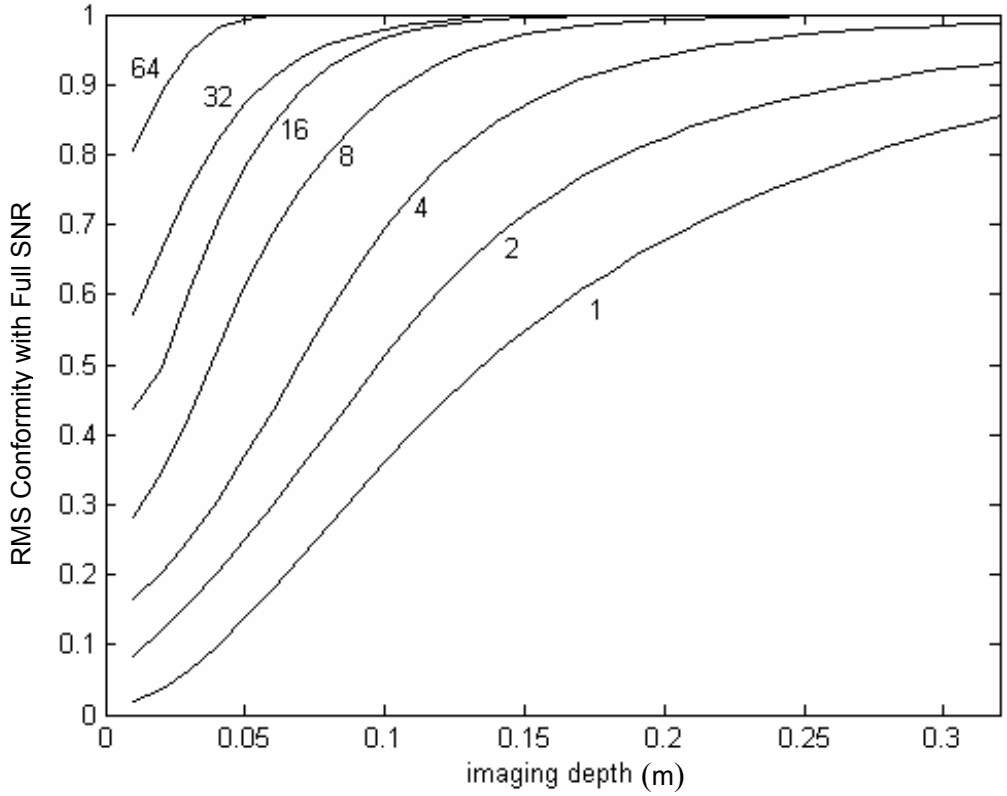


Figure 57. SNR performance of point combined 25cm square 9x9 grid array.

The normalized SNR map for a four channel point combine image is shown in Figure 58. At the chosen points, the SNR of this configuration is optimal, however between these points, there are significant nulls in the sensitivity of the combined array. The value of the deepest null is the maximum deviation from the optimal SNR map.

This value is plotted versus imaging depth for the various receiver configurations in Figure 59. Close to the array, all but the 64 channel configuration has a null that is at least 90% below optimal. The 64 channel combined array has a null that is 70% below optimal. As the distance is increased, the coverage of the SNR maps improve and the nulls become less deep. At an imaging depth of 10cm, using at least eight receivers ensured a maximum null that only reduces the SNR by 10%. The plot was repeated for the 25cm array in Figure 60.

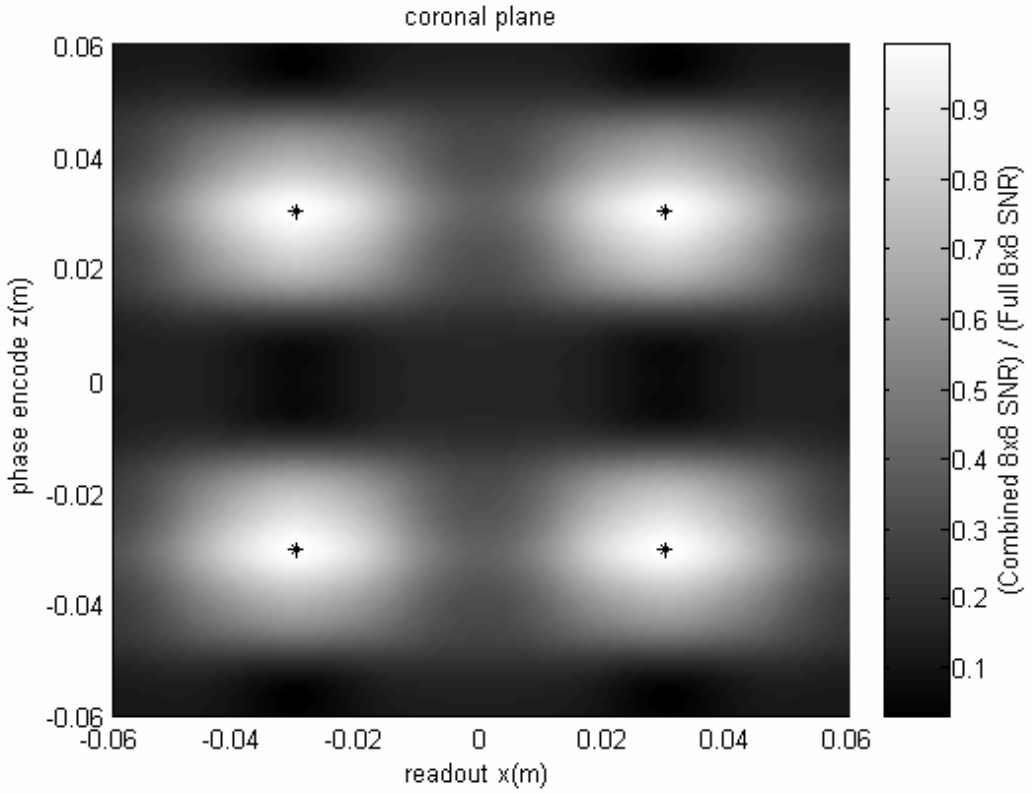


Figure 58. Nulls present in a point combined image.

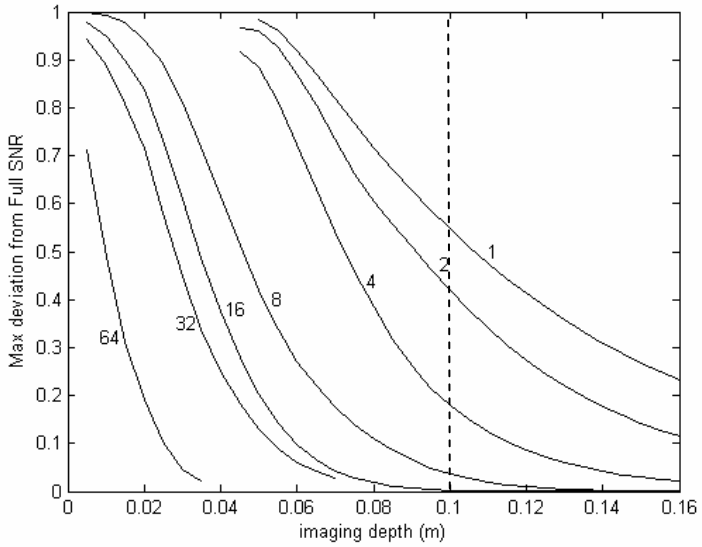


Figure 59. Maximum deviation versus depth for 12.5cm square array.

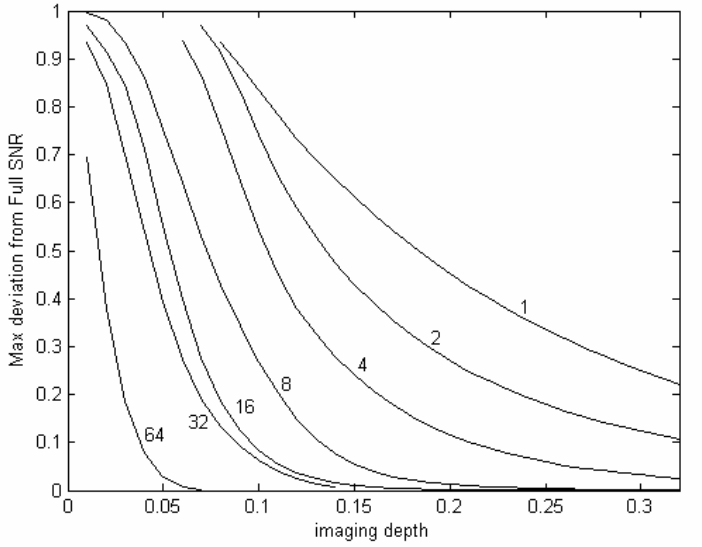


Figure 60. Maximum deviation versus imaging depth for 25cm square array.

Using a 64 channel linear array of planar pairs (21), fully encoded images were acquired using a 64 channel receiver (18). The images were processed in Matlab® where the k-space information was centered and a phase correction to account for the differences in phase offsets between the receiver channels. The data was then combined in software to simulate the effects of a hardwire combiner. The method for combination was to choose the weighting coefficients so that the signals are combined to yield optimal SNR at equally spaced points along the x-axis over the center of the array. The results are shown in Figure 61. The combined image using 64 receivers is shown in Figure 61a. As the number of receiver channels is reduced to 48, Figure 61b, 32, Figure 61c, and then 16, Figure 61d, the SNR decreases, but no major artifacts appear in the image. Banding appears to show up in the image due to the focusing of the array on the chosen points when 8 receivers are used, Figure 61e. The nulls in the array sensitivity become significant when only four receivers channels, Figure 61f, are used to combine the 64 elements. In this configuration, the image is reduced to bands and the detail of the sample is unrecognizable with only four vertical stripes.

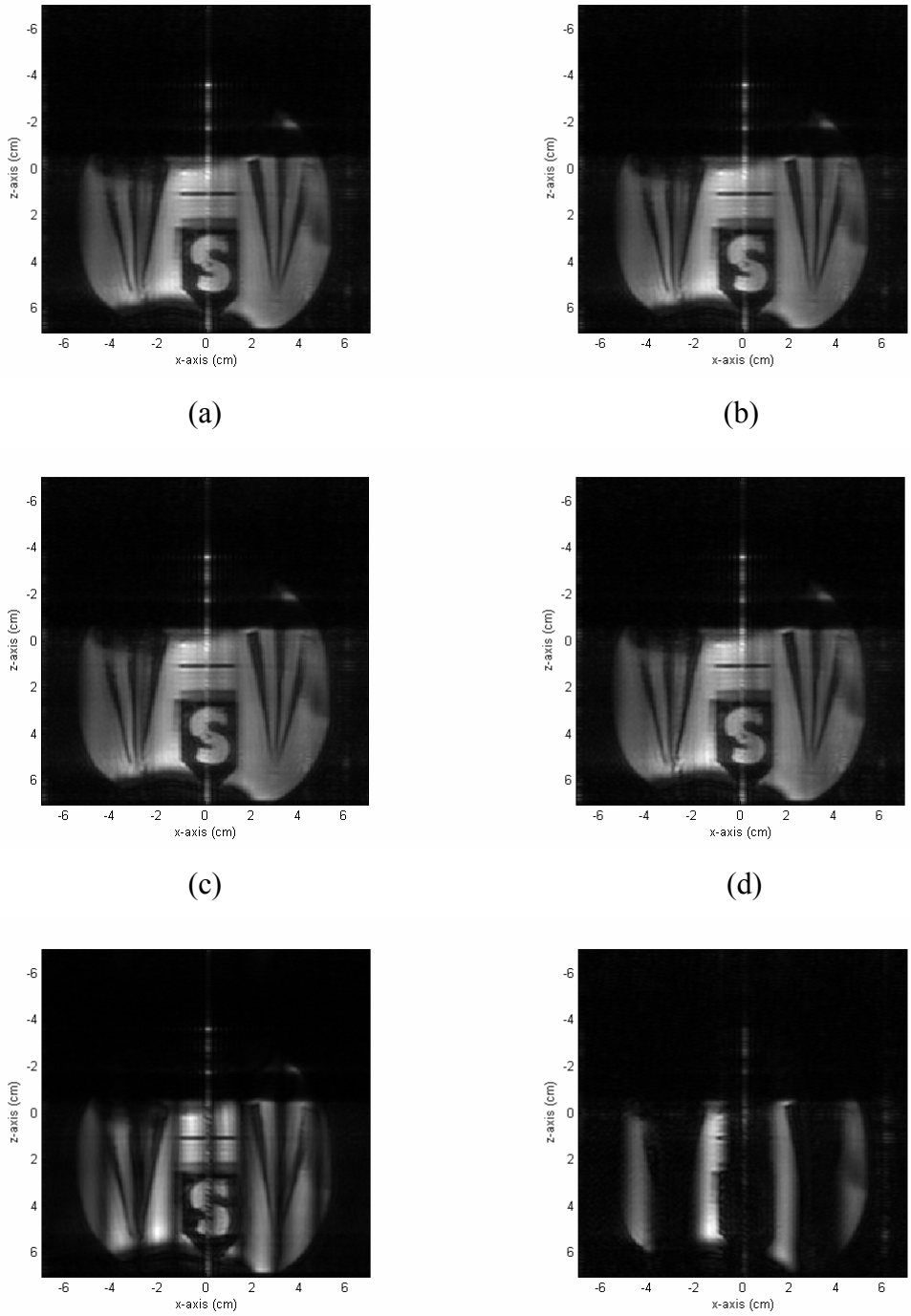


Figure 61. Point combined images at 1mm.

G-Factor

As discussed previously in Chapter II, the performance of an array during SENSE is quantified by the g-factor. Using Matlab®, g-factor maps and SENSE reconstructions were performed, using the 64 element planar pair array, for various combinations of numbers of receivers and reduction factors. The image field of view was 14cm by 14cm in the coronal plane parallel to a 12cm by 12cm array. The images were phase encoded in the x-direction, parallel to the short axis of the array elements, and frequency encoded in the z-direction.

The computed g-factor maps for the array are shown in Figure 62. For the point combined arrays, the g-factor results are very good. At a reduction factor of two, the g-factor is approximately equal to one over the entire FOV for any of the receiver choices. The more receivers of used, the more the g-factor is reduced. At a reduction factor of four, the g-factor shows maximum noise amplification of approximately .25% when only 8 receivers are used. For more receivers, the amount of noise amplification is negligible. For eight receivers operating at a reduction factor of eight, the g-factor performance is no longer tolerable. For this case, the maximum g-factor is 22 and translates into a noise increase of 2200%.

With sixteen receivers, the performance is much improved with the maximum g-factor is roughly 1.035. The 32 and 64 channel receivers perform similarly with a noise amplification slightly over 2%. At a reduction factor of sixteen, at least sixteen receivers are required. However, when only sixteen receivers are used at this reduction factor, the g-factor for the array is over 50. Therefore, more receivers should be used at this factor. With 32 receivers, the maximum g-factor is reduced to 1.75 and with 64 receivers, the maximum g-factor is slightly reduced to 1.7.

The SENSE reconstructions for the array using these receivers and reductions are displayed in Figure 63. At a reduction factor of one, the images are fully acquired and the SENSE algorithm optimally combines the channels to form the final image. With only eight receivers, obvious nulls in sensitivity can be seen in the final combined image. It appears that a minimum of sixteen channels are necessary, though this still results in noticeable banding in the final image when compared to the 32 or 64 channel images. All the different receivers perform well up to a reduction factor of four. The 16, 32, and 64 channel receivers perform well at a factor of eight as well. It is not until the reduction factor of 16 that noticeable blurring begins to appear in the final images.

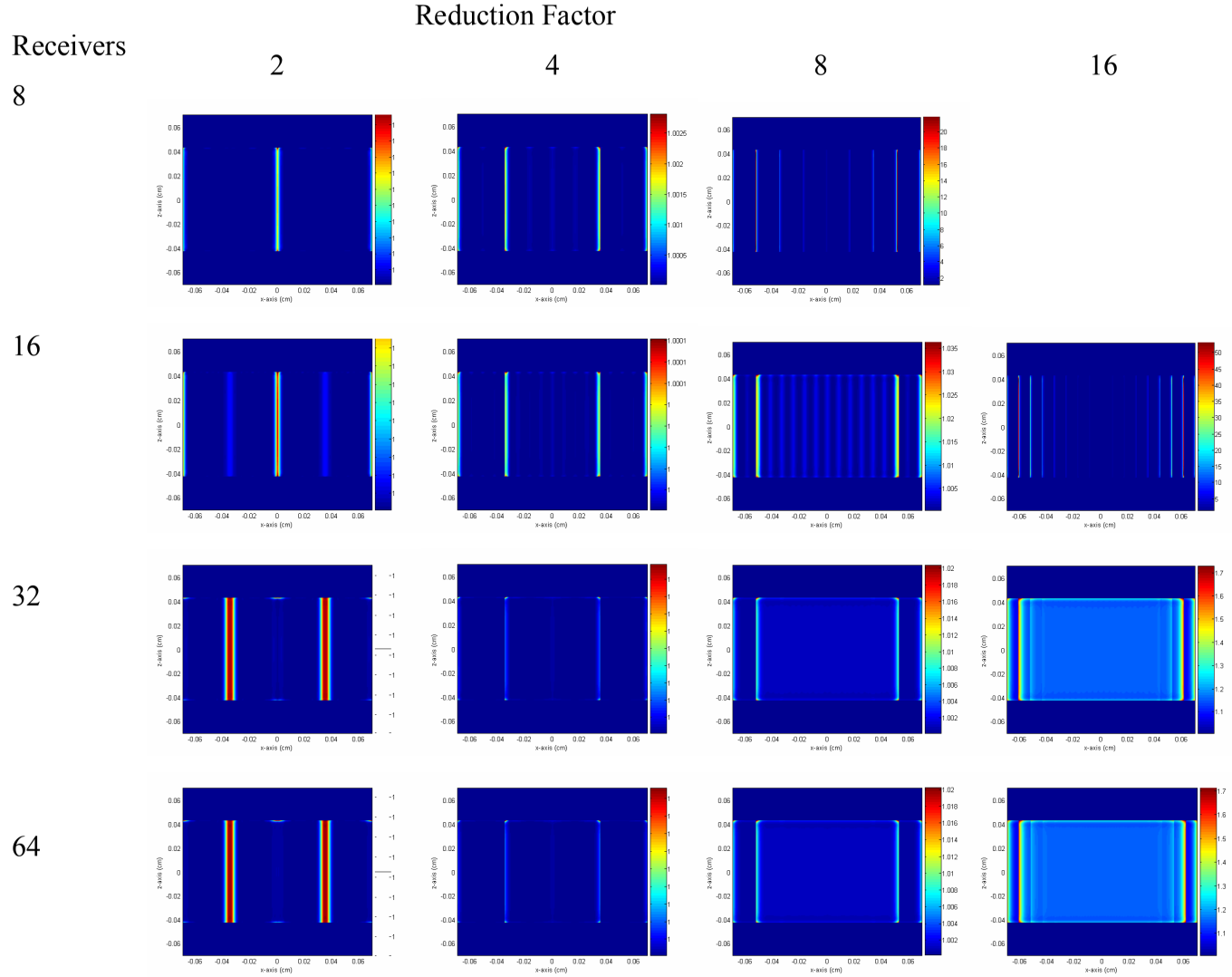


Figure 62. G-factor maps for point combined array.

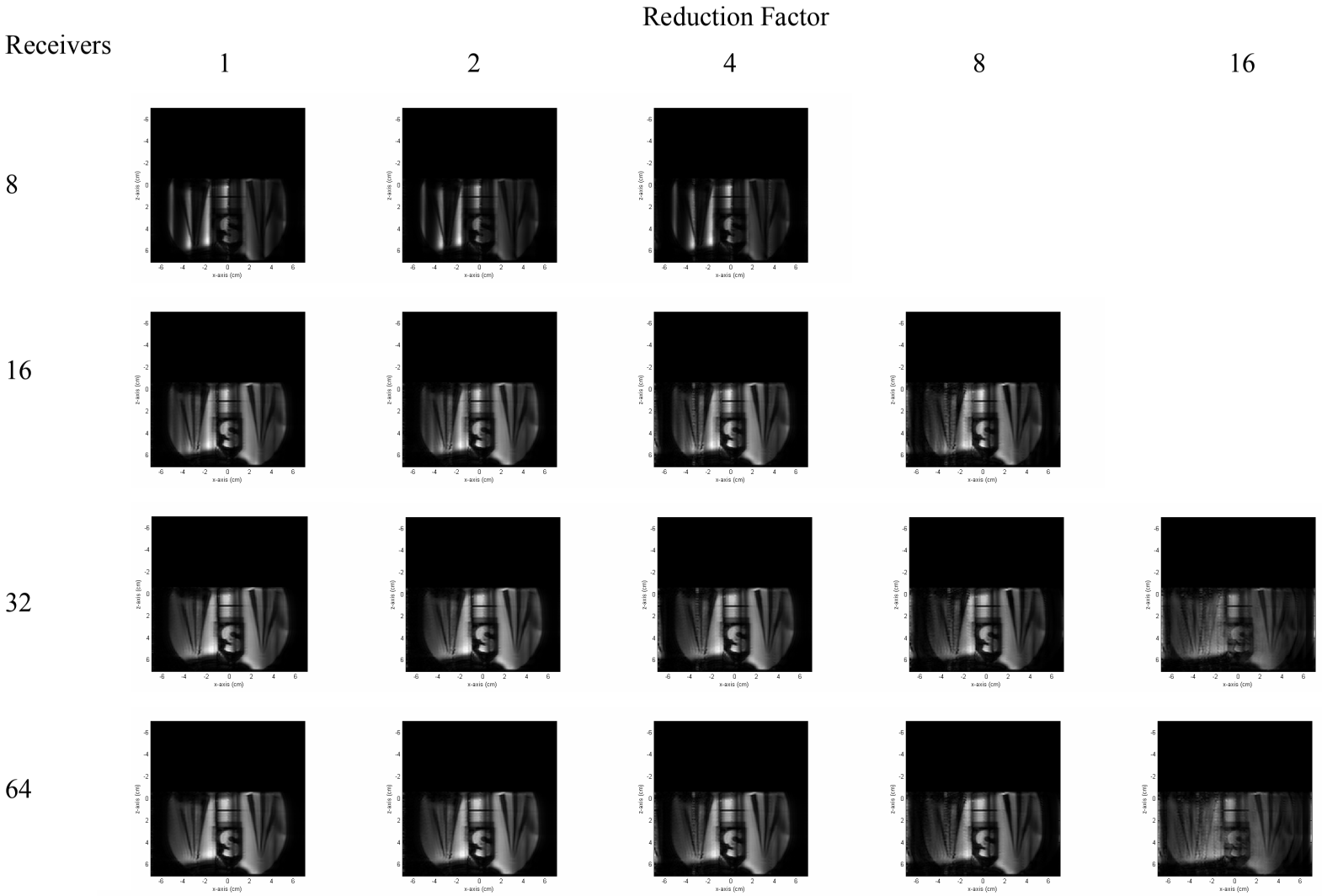


Figure 63. SENSE reconstructions using point combined array.

Region Combined Arrays

In order to eliminate the nulls in sensitivity caused by choosing weighting coefficients to optimize at points, Roemer's solution for finding the optimal weighting coefficients was extended in Chapter III to optimize the SNR of a region. In this section simulation results are computed and compared with the point combined arrays of the previous section. The geometry used for these simulations is shown below in Figure 64. Also in this section, data acquired for a 64 channel linear array is combined using region optimized weighting coefficients, and the results are discussed.

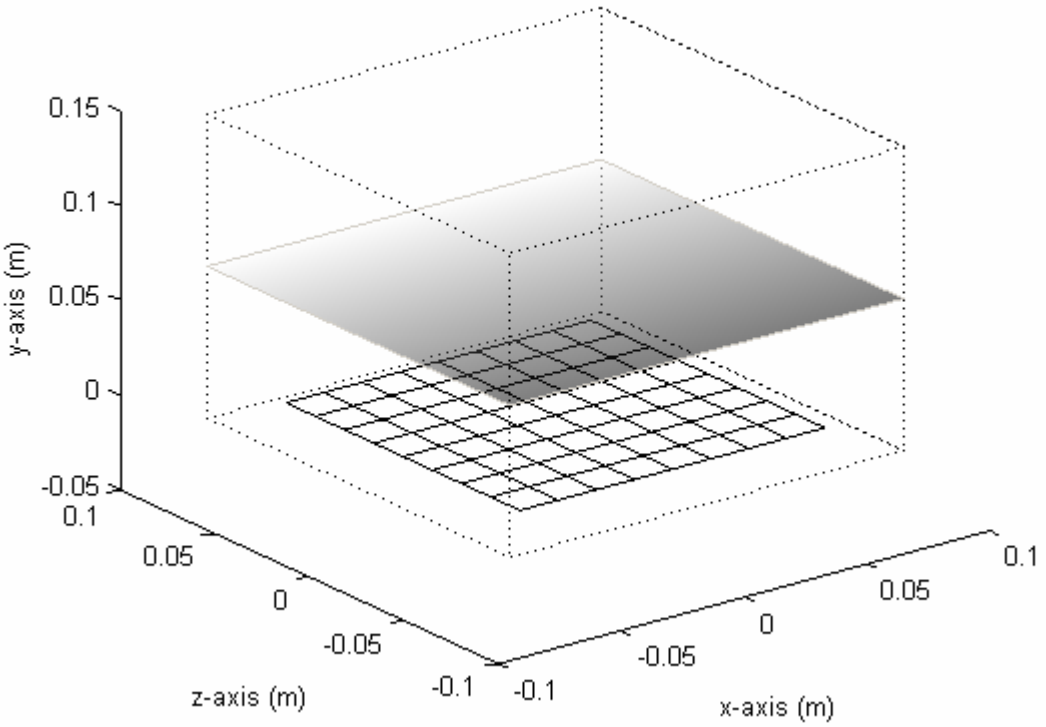


Figure 64. Grid array geometry used for simulation.

Signal to Noise Ratio

Using four receiver channels, the grid array was combined so that each channel would be responsible for one of the four quadrants of the image. A normalized SNR map of this arrangement is shown Figure 65 at a depth of 1.5cm. The asterisks show the centers of the four quadrants. The SNR over this field of view ranges between 20% and 80% of optimal. At a depth of 3cm, Figure 66, the SNR over the FOV ranges from 40% to 95% of the combined 81 element grid array.

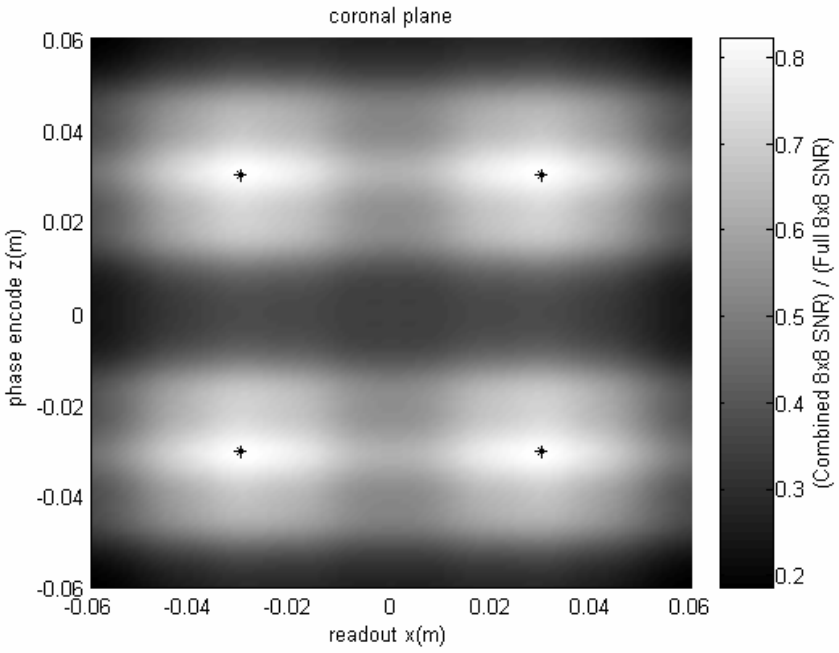


Figure 65. Region combined SNR map using four channels at 1.5 cm.

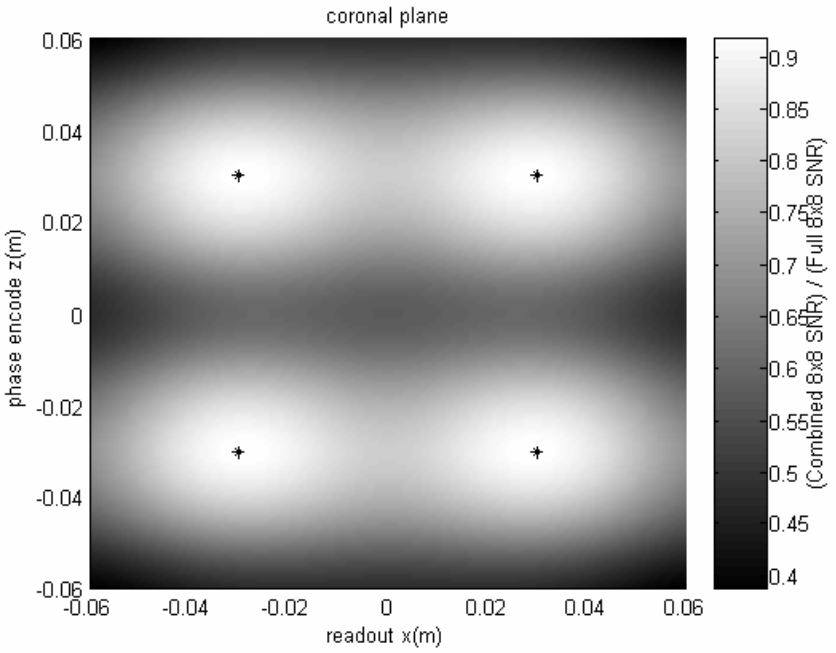


Figure 66. Region combined SNR map using four channels at 3 cm.

Comparing these results to the point optimized results of the previous section, significant improvement is seen in the coverage using the region optimized weighting coefficients. Maps of the improvement are shown in Figure 67 for a depth of 1.5 cm, and in Figure 68 for an imaging depth of 3cm. At 1.5cm, the point combined arrays have deep nulls that show considerable improvement using the region combined coefficients; an increase by a factor of 13 in some areas. At a depth of 3cm, the region combined arrays still show improvement in sensitivity over the point combined method. However, the gains are not as significant since the point combined sensitivities are already fairly diffuse due to the distance from the plane from the array. The maximum improvement of the region combined array at an imaging depth of 3cm is approximately a factor of two.

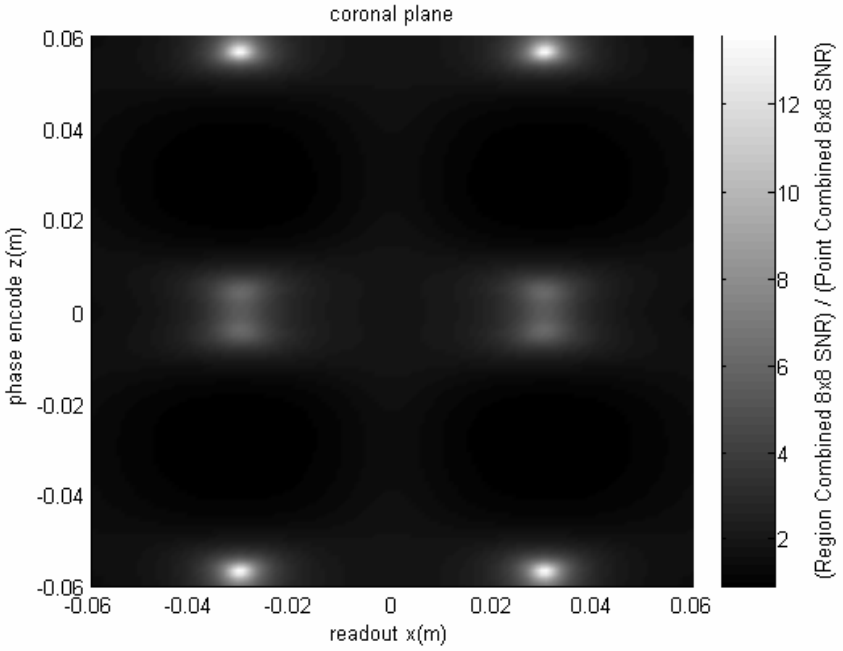


Figure 67. Ratio of point combined and region combined SNR maps at 1.5cm.

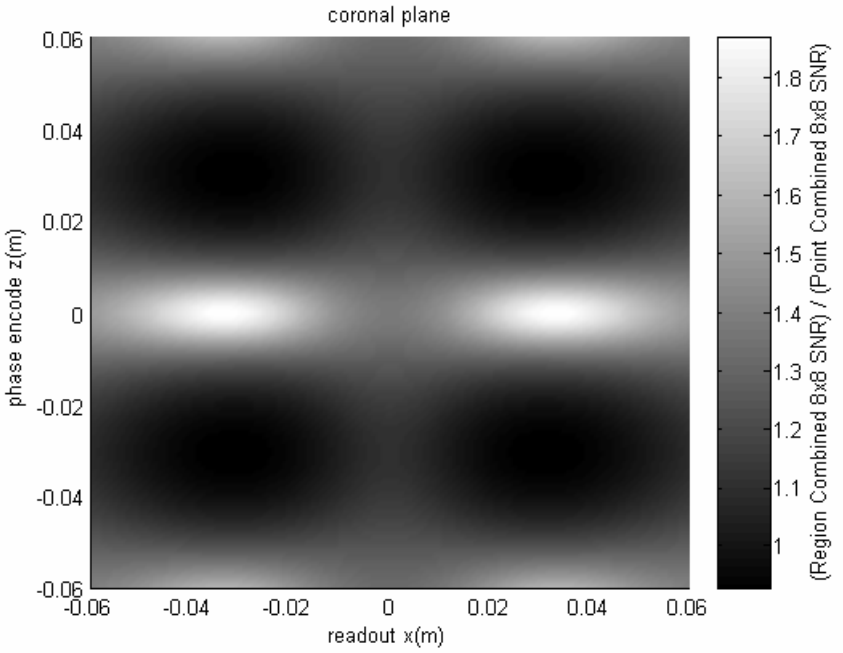


Figure 68. Ratio of point combined and region combined SNR maps at 3cm.

The deviation from the optimal SNR was computed for the regional coefficients .using Eq. [5.1], as the imaging depth was increased for one, two, and four channel configurations. A plot of the deviation is shown in Figure 69 where the point optimized curves were also shown for comparison. This plot shows the average improvement of the method over the entire FOV. For the single channel case, the region of optimization was the entire FOV. In the two channel situation, the FOV was subdivided and the SNR optimized over the upper and lower halves. And in the four channel setup, the field of view was divided into four quadrants. Near the surface of the array, the region optimized coefficients show approximately a 10% overall improvement over the point optimized method. Much of this improvement is due to the filling in of the nulls within the FOV. As the imaging depth is increased, the two methods converge.

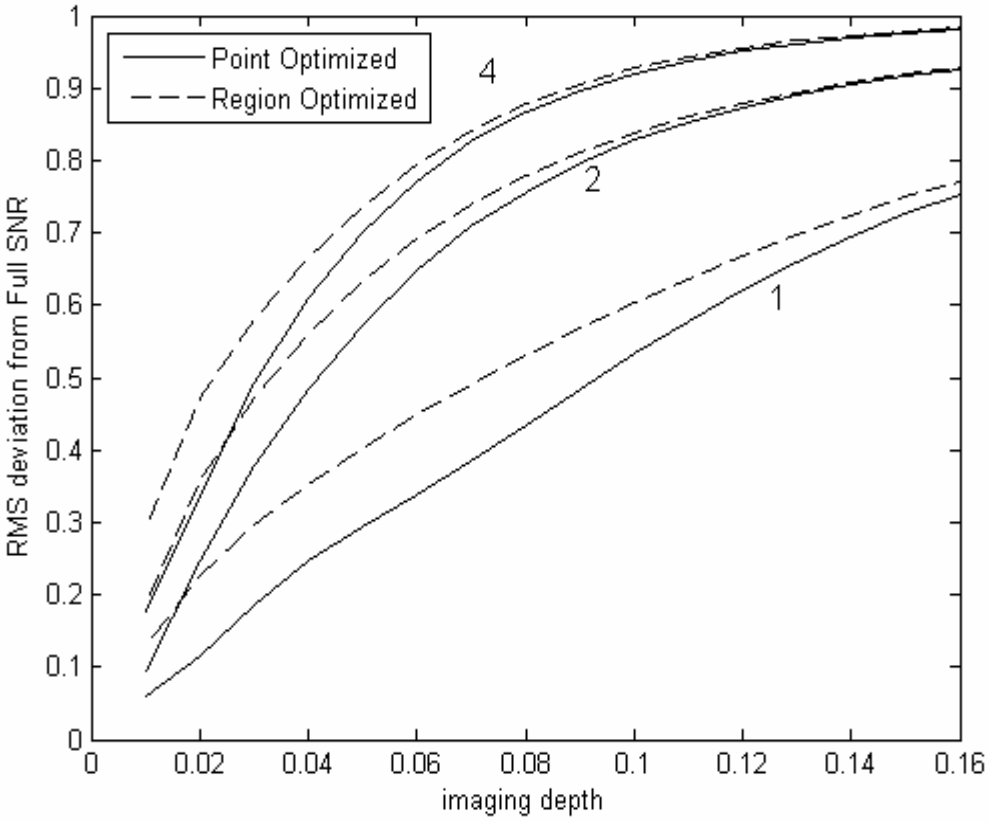


Figure 69. Comparison of point optimized and region optimized SNR versus depth.

In the previous region optimized simulations, the regions chosen for the for channels were adjacent. This is not required. In Figure 70, a normalized SNR for region optimized array at a depth of 1.5cm is shown using regions that now have a 2cm overlap. The length and width of the regions were extended so that the edges extend 1cm beyond the edges of the quadrant. Using the same regions, a normalized SNR map at an imaging depth of 3cm is shown in Figure 71. A comparison of these results to the point combined methods for the 1.5cm and 3cm is shown in Figure 72 and Figure 73 respectively. At 1.5cm imaging depth, the overlapped regions show approximately a factor of 14 maximum improvement in some regions of the image. At three centimeters, the patterns of the region combined and point combined sensitivities are more similar with the region combined shown a maximum factor of 2.2 improvement.

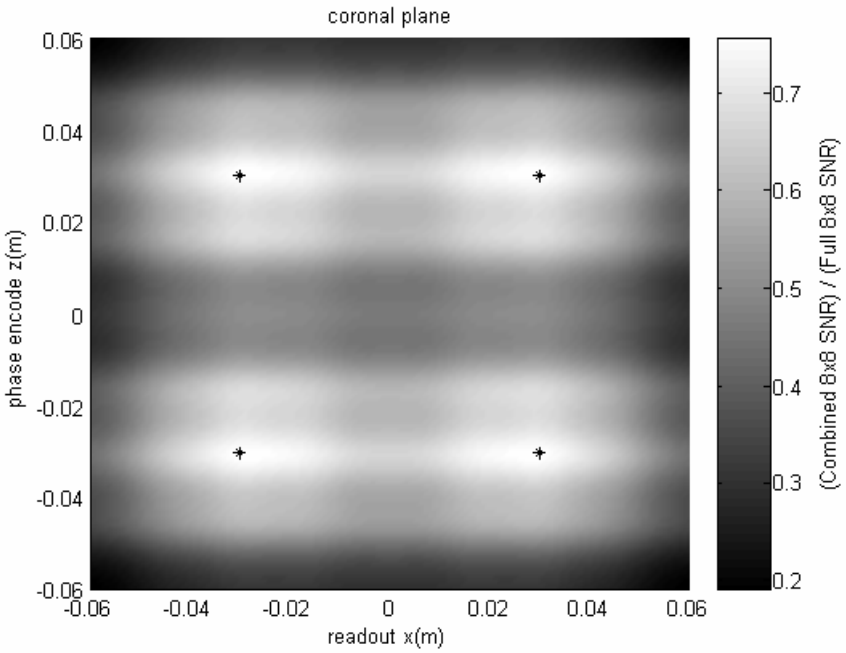


Figure 70. SNR map using overlapped regions at 1.5cm.

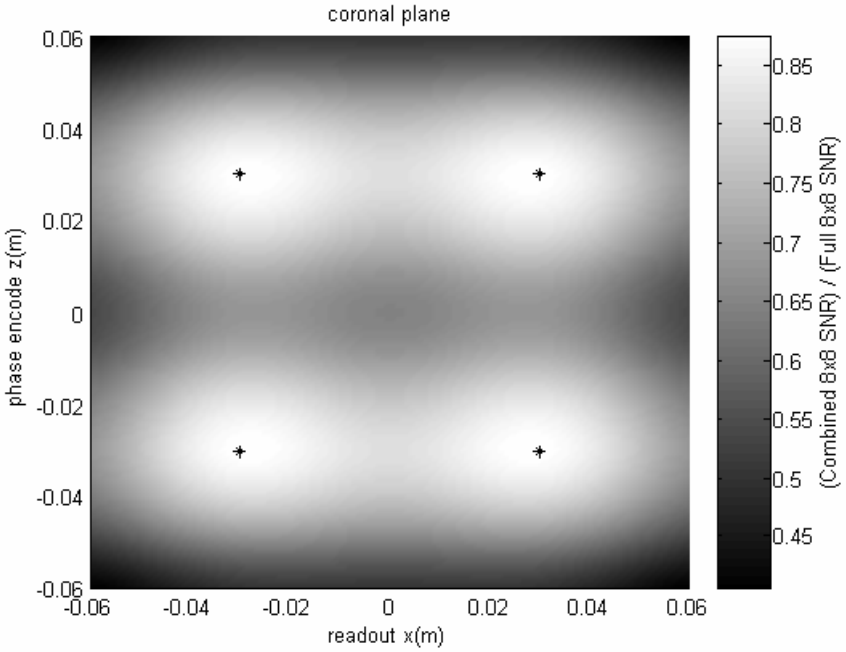


Figure 71. SNR map using overlapped regions at 3cm.

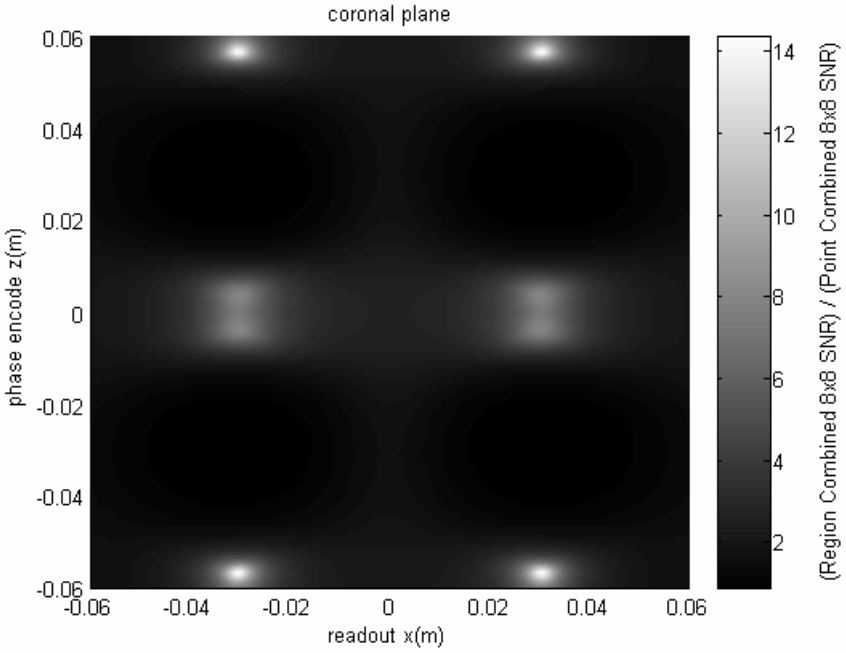


Figure 72. Comparison of overlapped region to point combined SNR at 1.5cm.

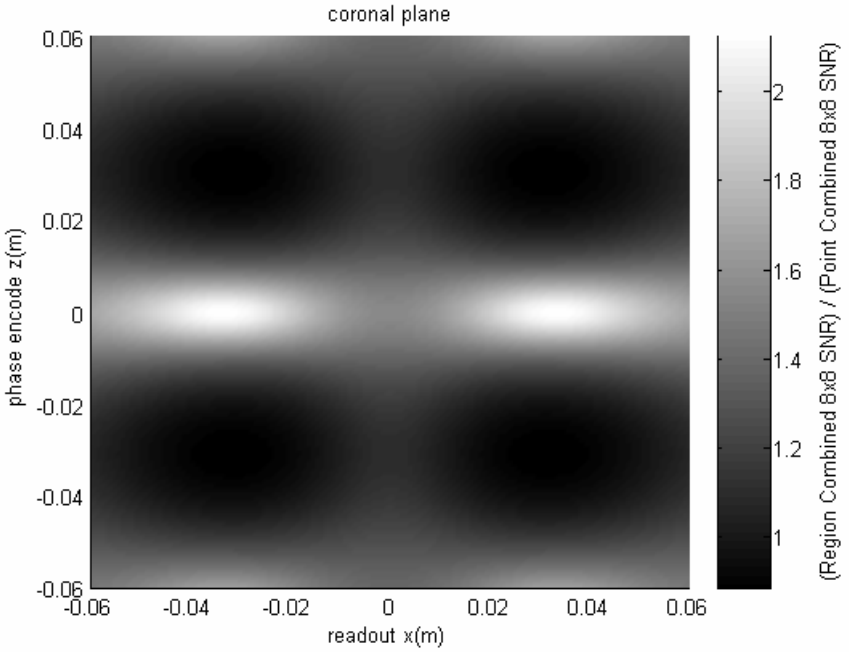


Figure 73. Comparison of overlapped region SNR to point combined SNR at 3 cm.

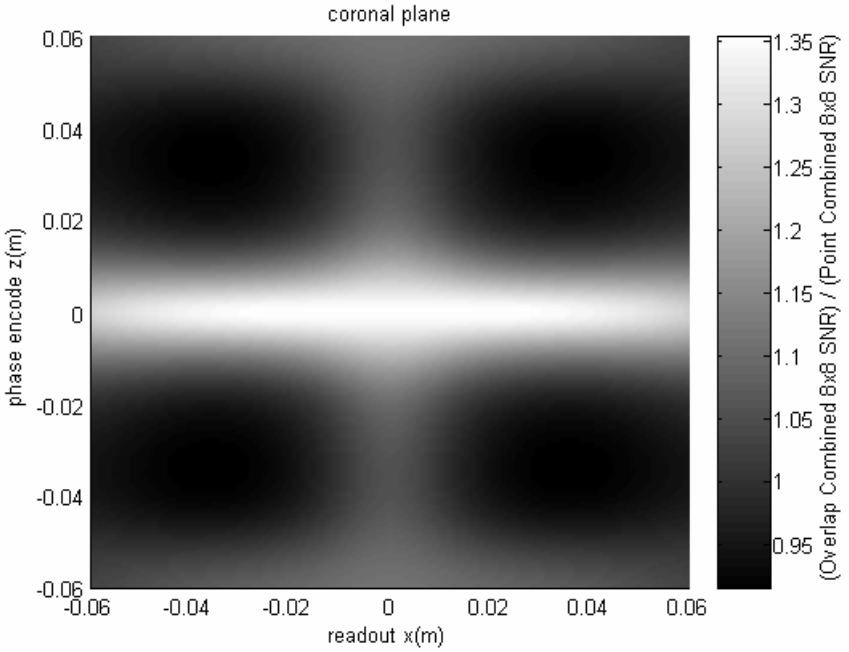


Figure 74. Comparison of overlapped region to region combined SNR at 1.5 cm.

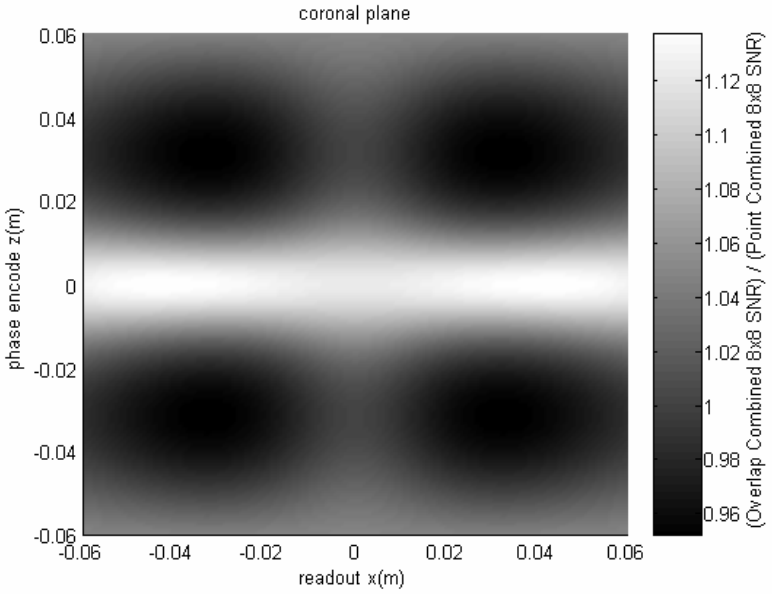
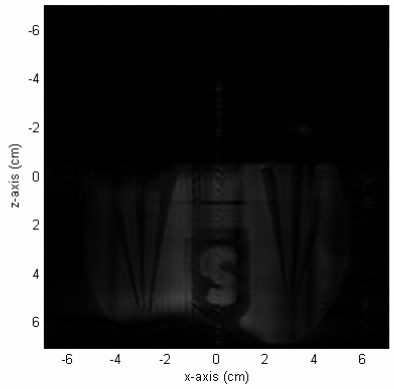


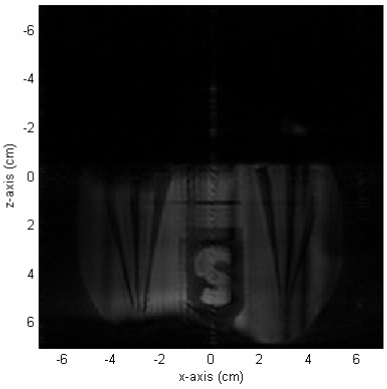
Figure 75. Comparison of overlapped region to region combined SNR at 3 cm.

Comparing the overlapped regions to the adjacent regions, Figure 74 and Figure 75, the tradeoff between the choices in weighting coefficients is highlighted. Using a 64 element linear array of planar pairs, complete image data sets were acquired and then combined in software to simulate the effects of combining an array to optimize SNR over a region. The 64 element array was combined into receivers having one, two, four, eight, or sixteen channels. The FOV was equally subdivided into a number of rectangles depending on how many channels were available. The long axis of the rectangles was parallel to the z-axis and had a length of 14cm. The short axis of the rectangle was in the phase encode, x, direction and had a length of 14cm divided by the number of channels.

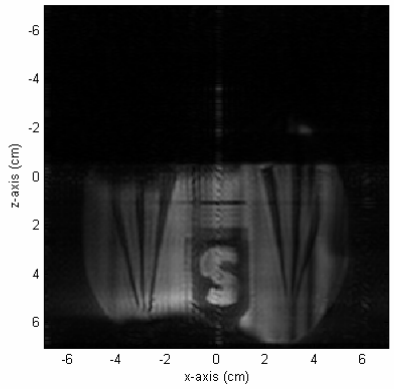
The region combined images are shown in Figure 76. The images have been normalized to the mean SNR of the sixteen channel combined image. The entire FOV is visible even for a receiver having a single channel, albeit with low relative SNR. As more channels are added, the SNR of the final composite image improves.



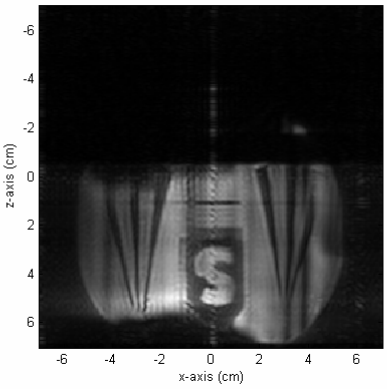
1 channel



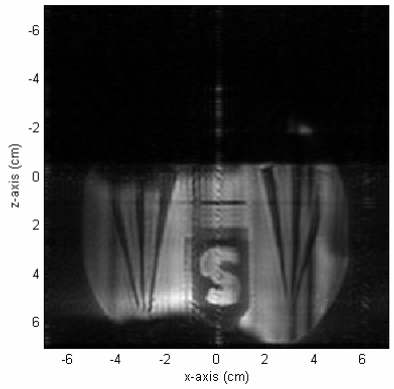
2 channels



4 channels



8 channels



16 channels

Figure 76. Region combined images using various receivers.

G-Factor

G-factor maps and SENSE reconstructions were processed for the 64 element linear array of planar pairs when the array was combined to optimize SNR over regions. The g-factor maps were computed for the coronal plane with a FOV of 14cm by 14cm. The FOV was subdivided into rectangles with their long axis perpendicular to the phase encode direction. The SNR was optimized over these rectangular regions. The number of rectangles used depended on the number of channels available in the receiver.

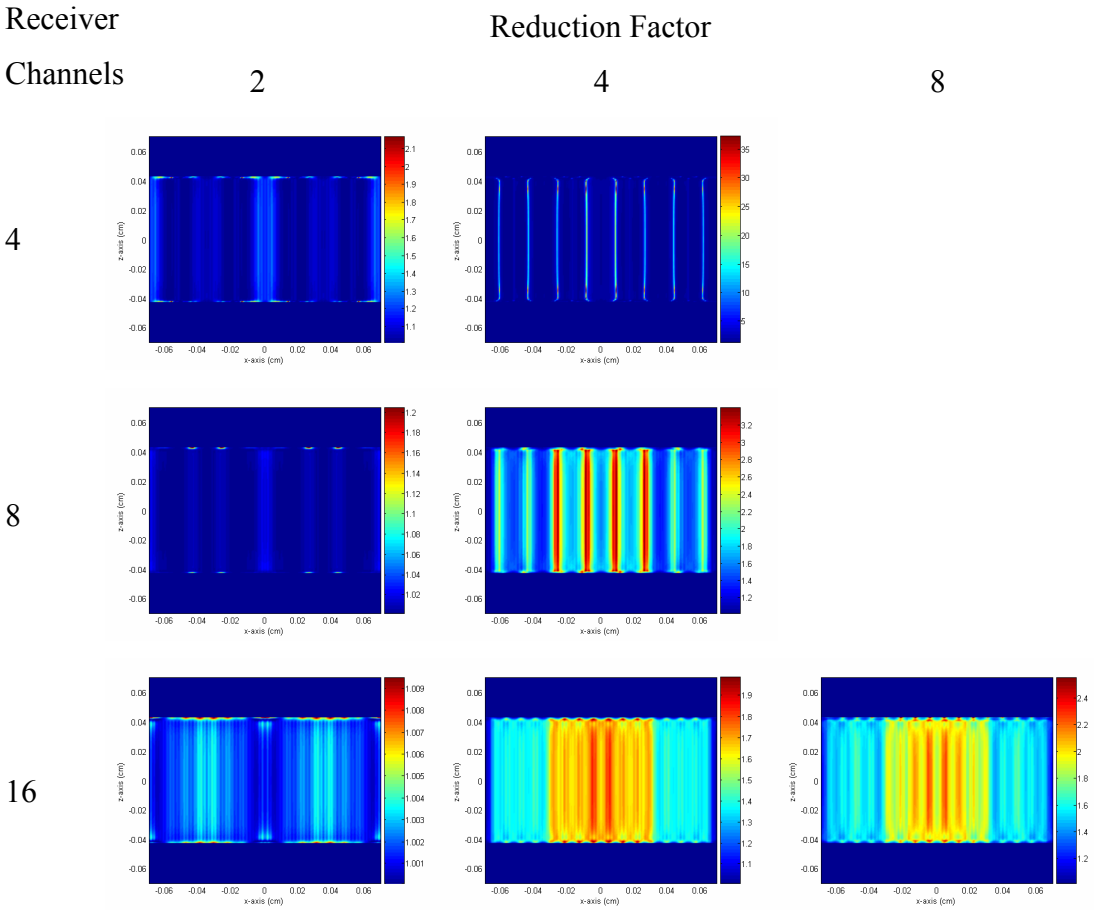


Figure 77. G-factor maps for region combined array.

The g-factor maps, using the planar pair array, are shown in Figure 77 for receivers have four, eight, and sixteen channels. At a reduction factor of two, the four channel receiver performs worst, as expected, with a maximum g-factor of 2.2.

Increasing the number of channels reduces the maximum value to 1.2 and 1.01 for eight and sixteen channels respectively. For a reduction factor of four, the g-factor performance is further reduced. At this reduction factor, the four channel array is now useless with a maximum g-factor greater than 35. Even the eight channel array shows very poor performance with g-factors of 3.3. The sixteen channel array is barely tolerable with g-factors less than 2. At a reduction factor of eight, the maximum g-factor for the sixteen channel receiver increases to 2.5.

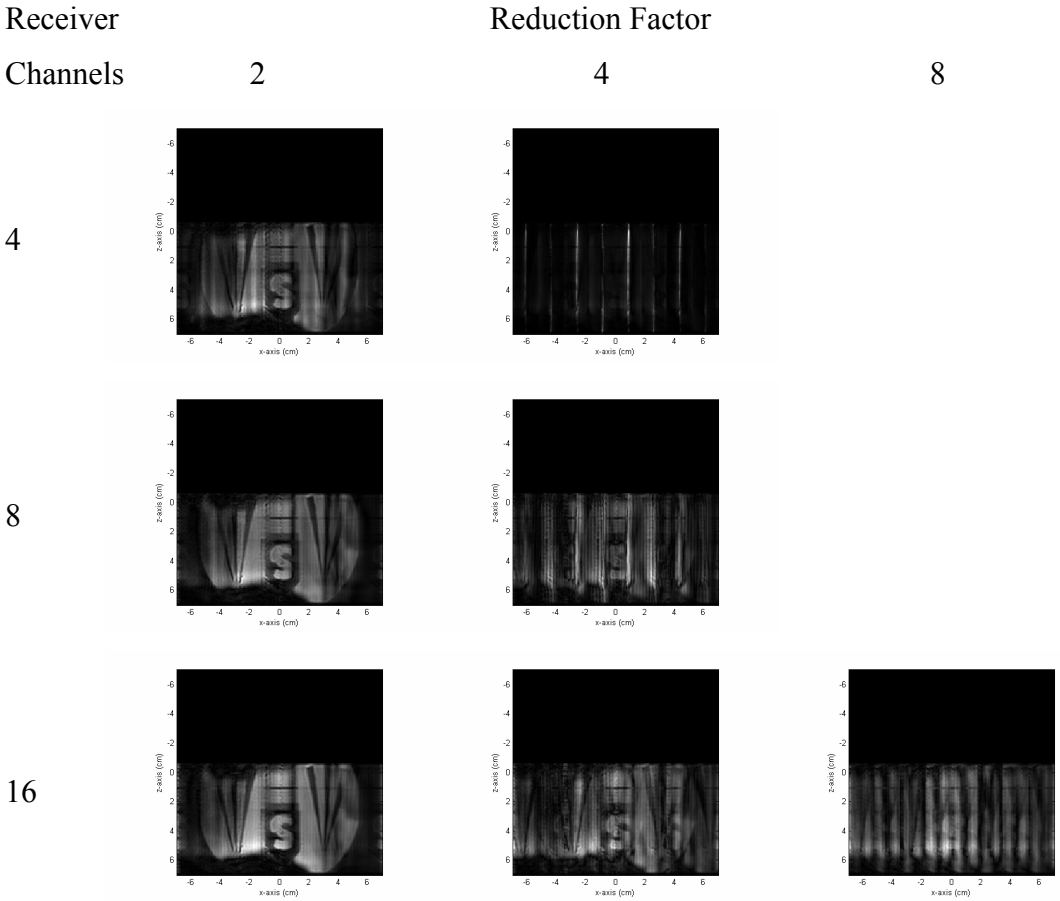


Figure 78. SENSE reconstructions using region combined array.

The SENSE reconstructed images using the linear array of planar pairs combined to optimize SNR over regions is shown above in Figure 78. The four channel receiver images show blurring at a reduction factor of two and completely obscured for a

reduction factor of four. The eight channel combined array improves upon the four channel situation slightly. The object is now discernable at a factor of two with slight smearing, but is still unrecognizable at a factor of four. The sixteen channel receiver performs well at a factor of two, but shows significant misregistration of the aliased pixels at factor four. At a factor of eight, the sixteen channel situation is unintelligible.

The poor performance of the region combined arrays for SENSE imaging is due to the sensitivity profiles of the combined channels. Whereas the point combined arrays had tightly focused sensitivity profiles, the region combined arrays are much broader. This results in significant overlap in sensitivity between the receiver channels and causes a loss of condition for the SENSE unfolding matrix.

G-Factor Optimized Arrays

Results were simulated for a 64 element linear array of loops combined to optimize g-factor. Calculations were done for arrays lying in the coronal plane with phase encoding in both the x and z directions. In these simulations, the 64 element array was combined down to four channels. G-factor maps and SENSE reconstructions were performed for reduction factors of two and four.

The functions used to fit the channel sensitivities are shown below in Figure 79. The functions are *rect* functions that have had a phase ramp applied to them. The width of the *rect* function corresponds to the FOV of the acquired image during maximum SENSE reduction. The amplitude of the sensitivity is equal to one over one quarter of the image FOV, while a phase ramp with a slope of -2, -1, 1, or 2 was applied to the respective channel. A phase offset between the channels was also enforced. The phase ramps work to improve performance at the transition from one region to the next where overlap in sensitivity is most likely to occur in the fitted profiles.

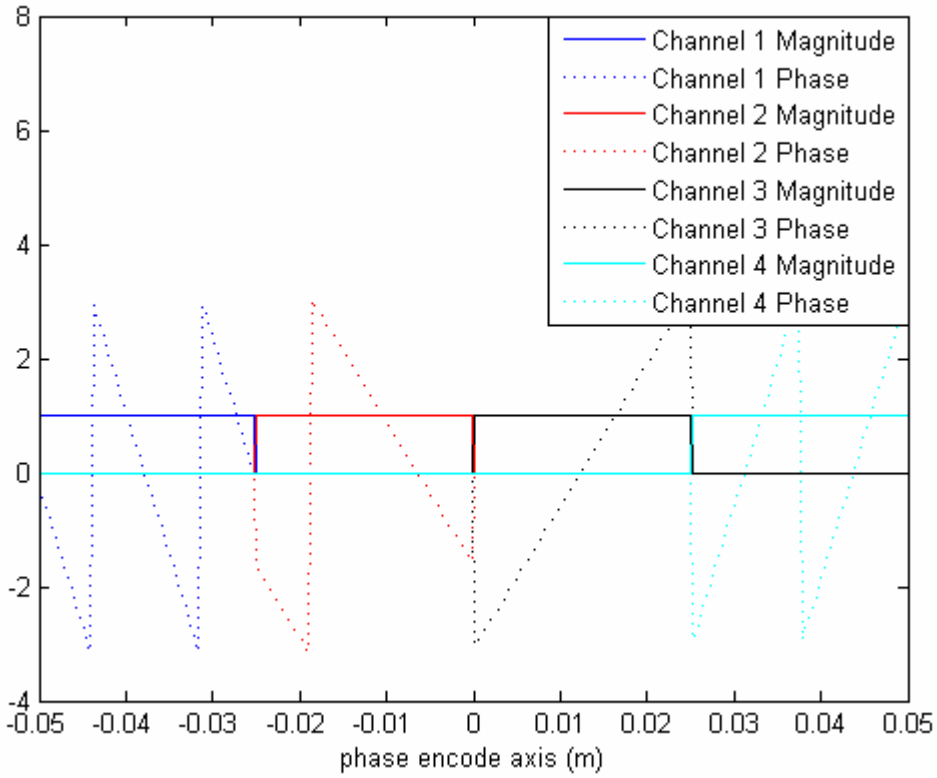


Figure 79. Forcing functions for four channel g-factor optimized combination.

When phase encoding in the z-direction, the long axis of the array element is in the x-direction while its short axis is in the z-direction. The effective magnetic field for this element is nearly entirely y-directed and provides a good basis set for fitting the forcing functions. The fitted profiles are shown in Figure 80, magnitude, and Figure 81, phase. The fits were done using 256 points on 4 lines for a total of 1024 test points. The actual profiles approximate the desired profiles closely. The magnitude plots who some ringing that occurs at the edges of the regions which results in some overlap at the transitions. The phase of the forcing functions has been fitted quite well, with some slight nonlinearity.

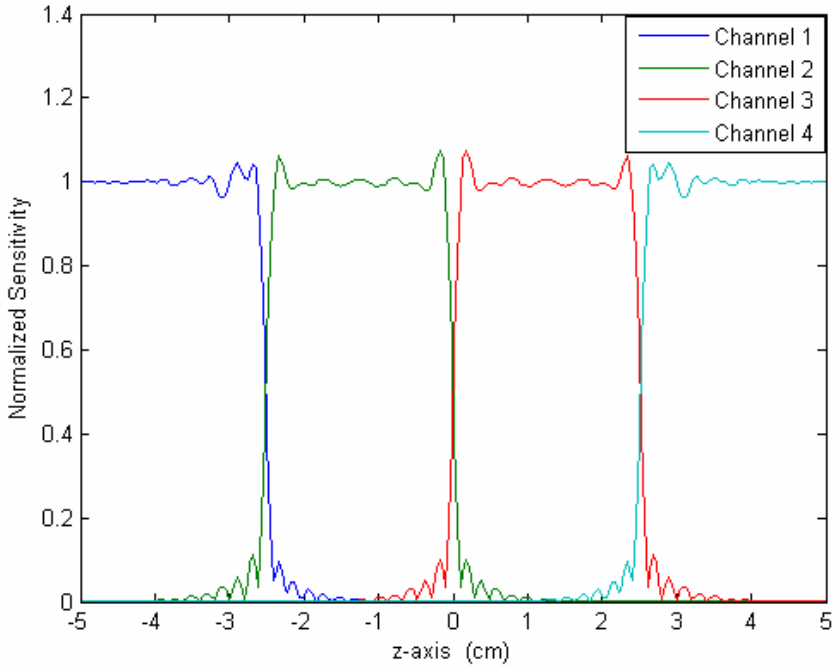


Figure 80. Fitted channel sensitivities magnitude for z-directed phase encode.

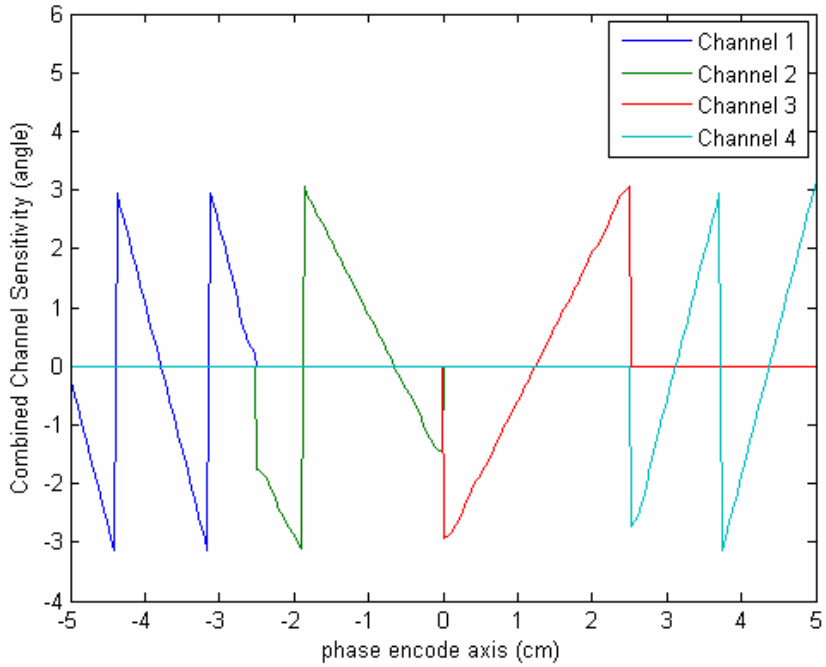


Figure 81. Fitted channel sensitivities phase for z-directed phase encode.

Using the fitted channel sensitivities, SENSE reconstructions were done on a simulated data set for reduction factors of one, two, and four. These SENSE images are shown in Figure 82. The square root of R loss of SNR between the images is immediately obvious. For a reduction factor of one, the image shows a loss of sensitivity in the transition between regions. When the acquired data sets are reduced, the images show an extra band in the center of the image, but there is no ghosting or blurring of the reconstructed image, even when the reduction factor is equal to the number of available channels.

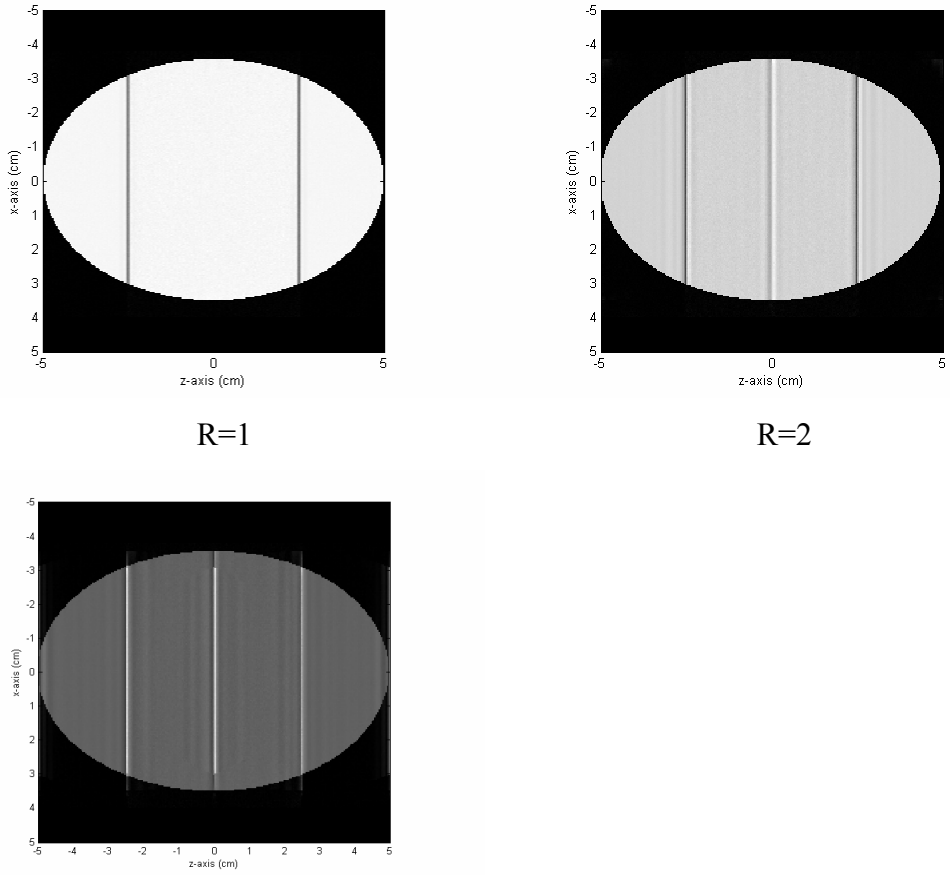
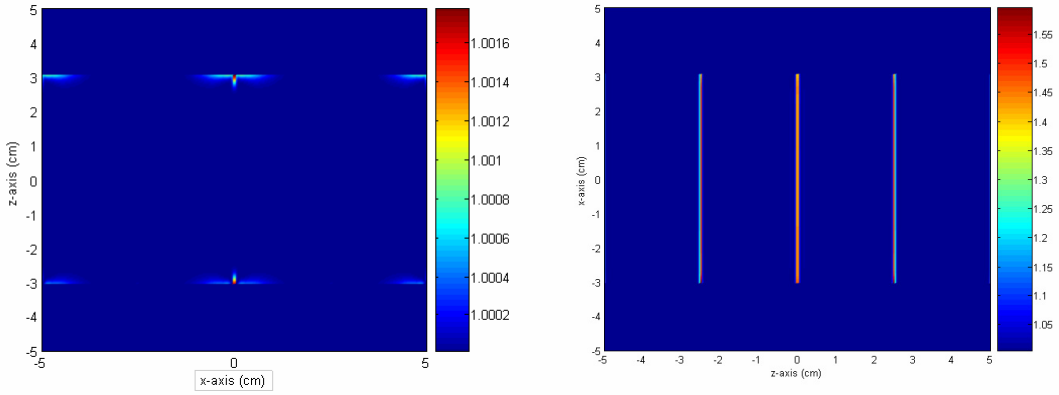


Figure 82. SENSE reconstruction of g-factor optimized arrays.

The g-factor maps for these reconstructions are show in Figure 83. At a reduction factor of two, the g-factor is very nearly equal to one over the entire image region; its maximum value is 1.0018. At a reduction factor of four, the maximum factor for a four channel receiver, the maximum g-factor is focused on three vertical lines that correspond to the transitions between the four regions. The maximum value for the g-factor is on the line in the center of the image as equals 1.6. Between these lines, the g-factor is approximately ideal with a g-factor very close to one.



R=2 R=4

Figure 83. G-factor maps for arrays combined for optimum g-factor in z-direction.

The situation becomes more complicated when phase encoding in the x or y directions because the array elements having large real and imaginary components. The more complex basis functions are not as easily fit to the desired forcing functions. When the complex forcing functions, those used previously for the z-directed phase encode, the fitted solutions fail to converge. Simplifying the functions to be simple *rect* functions with no phase variation, the weighting coefficients are solved for. The resulting channel sensitivity profiles are shown in Figure 84. These profiles exhibit a significant amount of error compared to the z-directed case. The profiles are not very smooth and have significant crossover into adjacent regions.

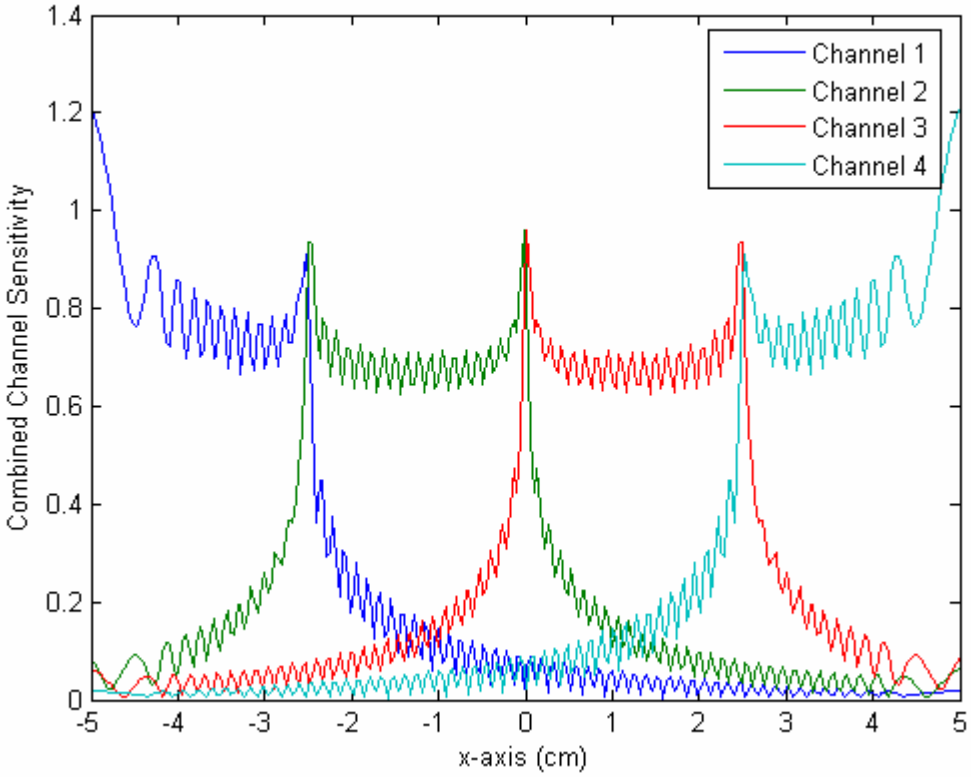
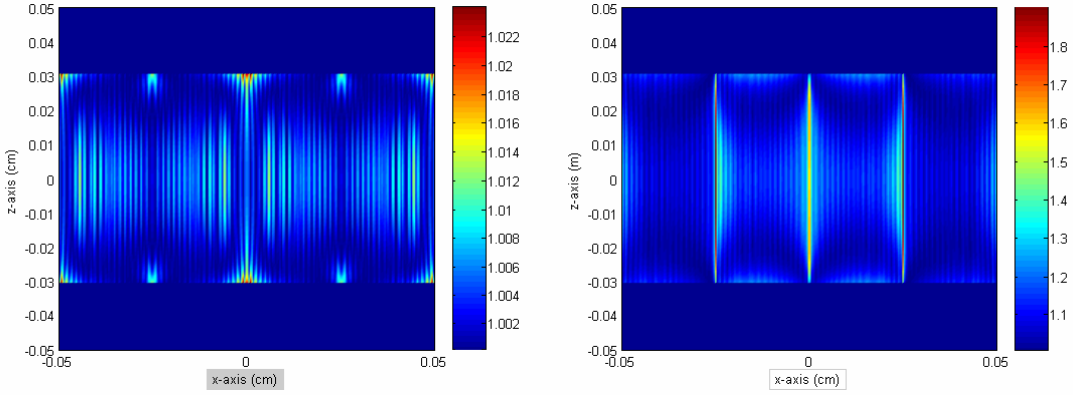


Figure 84. Fitted channel sensitivities for x-directed phase encode.

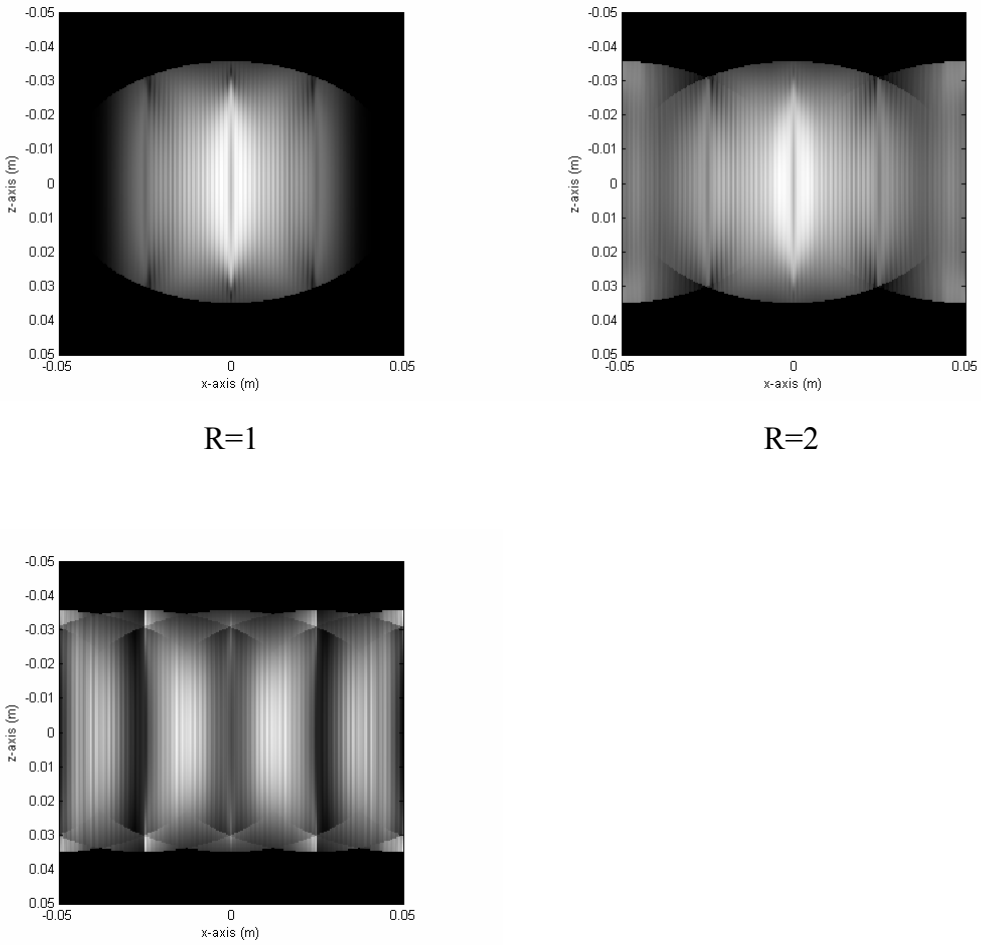
Computing the g-factor maps for the image using these fitted profiles, Figure 85, the maximum noise amplification is seen to be 1.024 reduction factor of two, and 2.0 for a reduction factor four. At both reduction factors, the g-factor maps are seen to not be very smooth. The jaggedness of the sensitivity profiles creates ridges in the g-factor maps.



R=2 R=4

Figure 85. G-factor maps for arrays combined for optimum g-factor in x-direction.

Performing the SENSE reconstruction using these fitted profiles, Figure 86, the lack of uniformity in the sensitivity profiles can immediately be seen. When the reduction factor is increased, ghosting can be seen in the reconstructed images. This is due to the strong overlap in sensitivities between adjacent channels. In order to phase encode in the x or y directions, a better set of forcing functions are needed. How to find these functions is left for future work.



R=1 R=2

R=4

Figure 86. SENSE reconstruction of g-factor optimized arrays.

CHAPTER VI

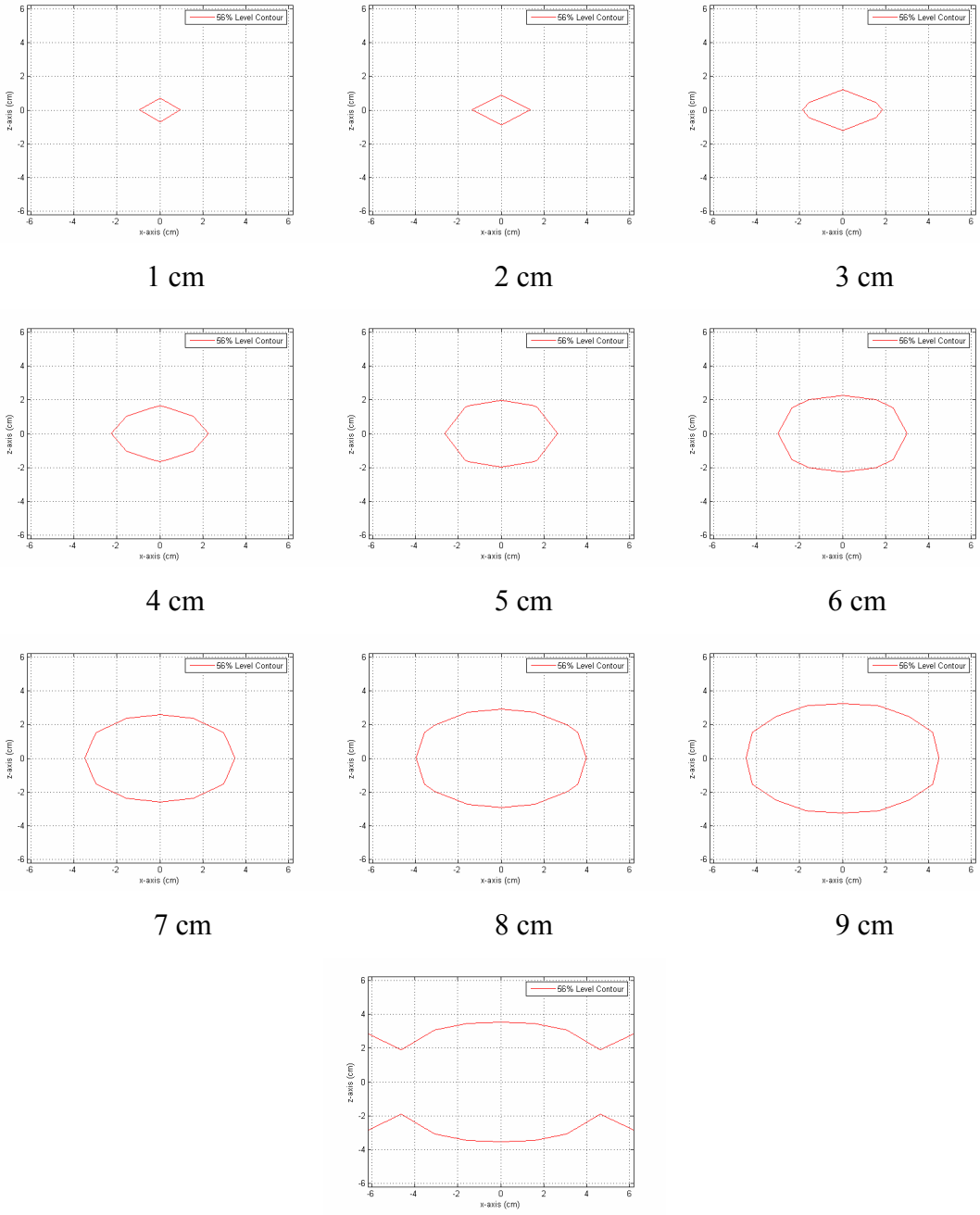
DISCUSSION

A set of tools and methods have been developed and implemented for this dissertation that can be used to examine questions relating to parallel imaging. In this chapter, the questions of how many receivers are required for parallel imaging and how to minimize noise amplification are discussed. This chapter also includes an overview of how the hardware combiner might be constructed, with a focus on dynamic range and resolution requirements for implementing the weighting coefficients.

Number of Receivers Required

As the number of receivers available in MRI scanners has increased, the question of how many receivers are actually necessary has arisen. Conventional wisdom has stated that more channels is generally better, but at some point there must be diminishing returns. The models created for this dissertation allow this question to be analyzed and some generalizations made.

By examining the weighting coefficients required to create the optimal SNR at a point centered on a 9x9 grid array, the resulting current sheet can be related to an effective coil radius. Figure 87 shows the effective radii for the current sheet as the imaging depth is increased from one to ten centimeters. The elliptical contour corresponds to the 55% of max value for the current sheet. This level was chosen since effective radius of the major axis is equal to the radius of a wire loop that optimizes SNR at the given depth. The contour is elliptical, with a 4/3 aspect ratio, due to the insensitivity of the $\hat{\mathbf{z}}$-component of the array to the MR signal. The effective coil created by the current sheet increase roughly linearly in size with imaging depth until the major diameter of the ellipse approaches the overall dimensions of the array, Figure 88. At this point the currents on the edge of the array begin to increase dramatically in order to compensate.



1 cm 2 cm 3 cm

4 cm 5 cm 6 cm

7 cm 8 cm 9 cm

10 cm

Figure 87. Effective coil radius as imaging depth is increased.

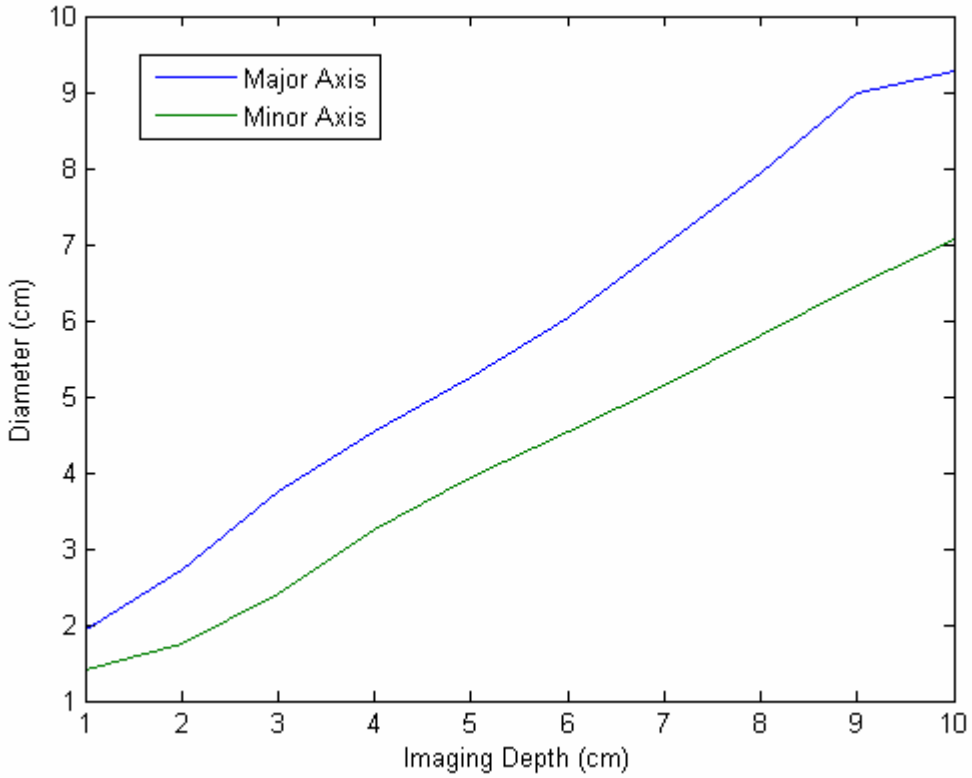


Figure 88. Diameter of major and minor axes of effective coil.

Based on the dimensions of an effective coil for a given coil sheet, the number of receivers required for a given imaging depth would be equal to the number of effective coils needed to maximize the coverage of the FOV, assuming the array dimensions are larger than the FOV. A quick approximation can be found by dividing the area of the image FOV by the area of the effective coil and is given by

$$\#Receivers = \frac{4lw}{\pi ab} \qquad [6.1]$$

where l and w are the length and width of the image FOV and a and b are the lengths of the major and minor axes of the effective coil. This approximation is valid when the effective coil dimensions are larger than the elements in the base array and smaller than the overall array size. Close to the array, the effective radius is small and many channels are required to fill the imaging plane. As the distance increases, the effective radius

increases and fewer channels are required until the area of the effect coil approximately equals the array aperture and only one channel is required. This approximation is also only valid for estimating the number of channels for maximizing SNR.

G-Factor

Combining the array elements to fewer channels for SENSE imaging is more complicated. Not only must SNR be maximized throughout the image, but g-factor must be minimized. This can be a difficult trade off. In the previous chapter, it was shown that the SENSE reconstructions for point combined arrays performed much better than region combined arrays in terms of g-factor at high reduction factors due to the sensitivity patterns being focused on the chosen points. The point combination method creates a set of basis functions that are highly orthogonal and result in exceptional g-factor performance as was shown in the previous chapter. However, this g-factor performance came at the cost of SNR. The point combined image SNR maps showed unacceptable deep nulls in the sensitivity.

In order to maintain SNR while minimizing g-factor, the array combinations need to work to orthogonalize the channel sensitivities while maintaining uniform coverage of the image FOV. The method of fitting the array sensitivities to an ideal profile was demonstrated in this dissertation. For an array constructed to phase encode in the z-direction, the fitting method worked very well and the channel sensitivity profiles closely approximated the desired profiles. This was due to the effective magnetic field of the array elements being nearly completely y-directed. For elements that hand more complex effective magnetic field patterns, the fitting method did not work well when trying to approximate the *rect* functions. Other functions, such as a Gaussian, may work better in this situation. Finding these functions is left to future work.

Hardware Combination

The combination of signals in hardware is mathematically identical to combining the signals in software,

$$s(t) = \sum_{c=1}^{N_c} w_c a_c(t). \qquad [6.2]$$

However, care must be taken in the design and implementation of the hardware combiner so that reduction in SNR does not occur. Figure 89 shows the schematic of a typical MR receiver channel. Immediately following the matching network for the coil is a high gain/low noise pre-amplifier which establishes the noise figure of the receiver. This is usually followed by an image rejection filter and then a secondary gain stage which amplifies the signal to a level so that it can be demodulated to an intermediate frequency. At the intermediate frequency, the signal is amplified again using a variable gain amplifier, so that the signal level can be user-controlled, and is then split and mixed down to baseband using a quadrature demodulator. The output of the demodulator has two signals that represent the real and imaginary portion of the MR signal. These signals are amplified once again, so that the full dynamic range of the digitizer is utilized, then filtered to the bandwidth of the MR signal, and finally digitized. This receiver architecture is typically repeated for every coil in the array and is independent for every channel, Figure 90.

When signals are combined, a combination network is inserted in the receiver chain that ties all the receiver channels together. The combination network can be inserted anywhere in the receiver chain. When used as for decoupling channels, the combiner is usually inserted before the pre-amps and is referred to as a decoupling network. For large arrays, this network can be quite large and complex and the assumption that it is

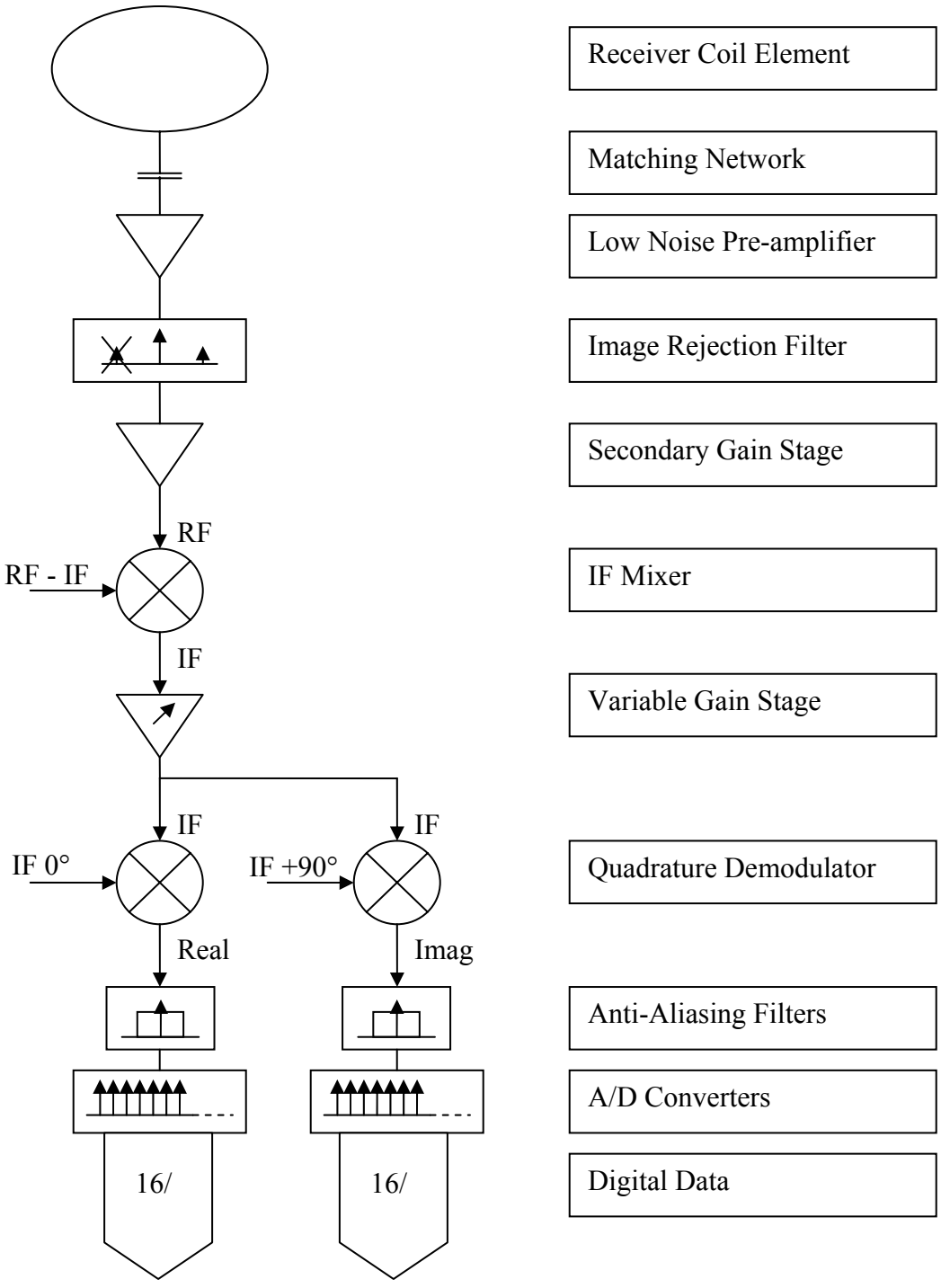


Figure 89. Typical MR receiver channel.

lossless may no longer be valid. In this case, the noise figure of the network will dominate the noise figure of the entire receiver.

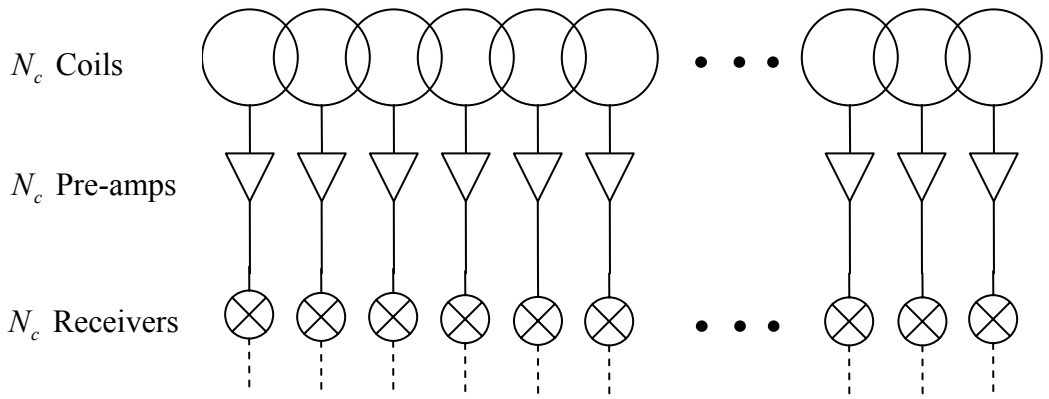


Figure 90. Block diagram of receiver array with independent channels.

To ensure that the noise figure of the receiver is dominated by the low-noise pre-amplifiers, it is best to insert the combination network after the pre-amplifiers. Since the pre-amplifiers are typically close to the coils and the rest of the receiver is located in a chassis, often in another room, it makes sense to insert the combiner network at this point, Figure 91.

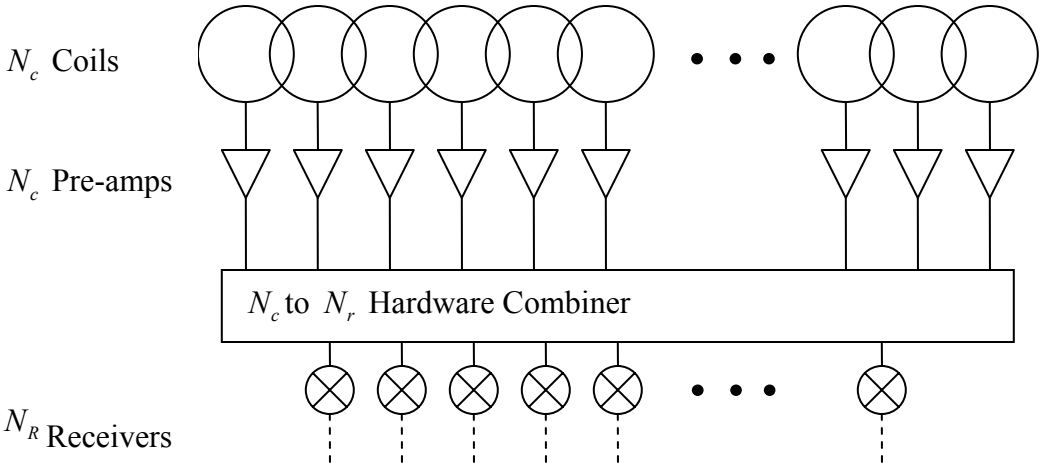


Figure 91. Block diagram of receiver array including hrdware combiner.

The combination network itself is designed to add the signals from the coil array according to a set of weighting coefficients. A block diagram of the combination network, when combining for a signal channel, is shown below in Figure 92. In both cases, the isolation between channels should be high so that there isn't any undesired coupling between channels due to the combiner.

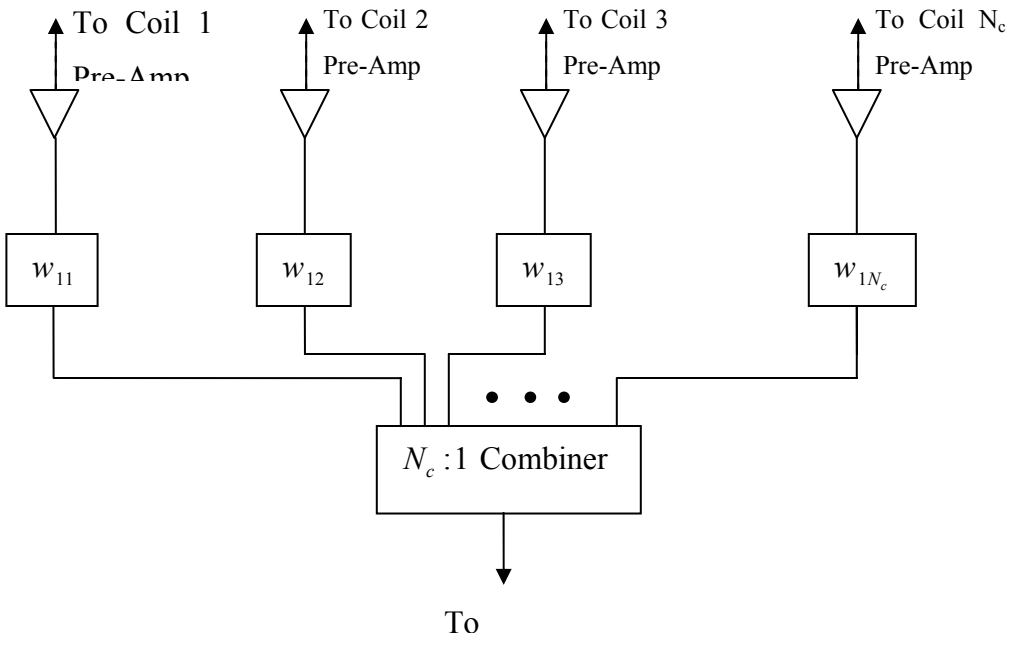


Figure 92. Block diagram of hardware combiner for a single receiver channel.

When combining an array for multiple receiver channels, the network is expanded into a matrix as shown in Figure 93. Components for splitting and recombining large numbers of signals are commercially available. However the variable gain amplifier and variable phase shifter will still need to be added. A combiner network with N_c coils and N_r receiver channels will require $N_c \cdot N_r$ attenuators and phase shifters in order to handle every possible choice of weighting coefficient.

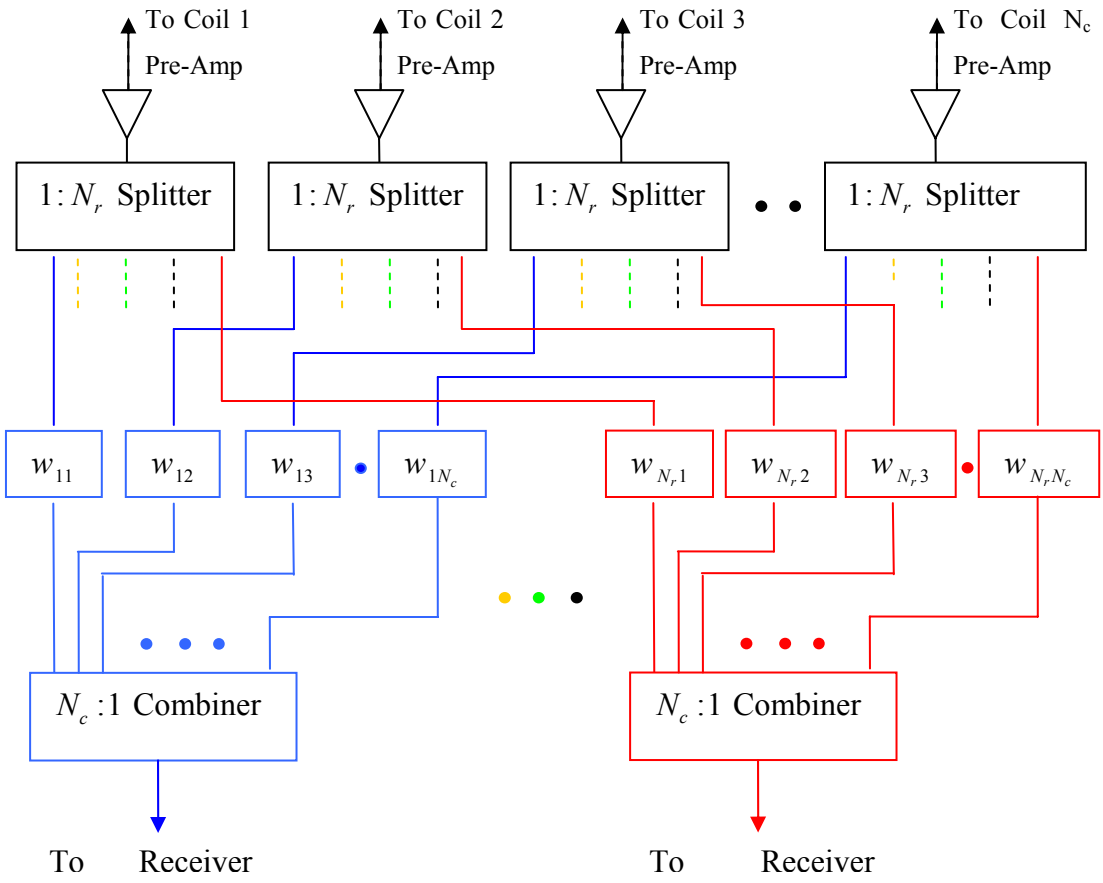


Figure 93. Block diagram of hardware combiner with multiple receivers.

The values for the weighting coefficients can either be implemented in hardware in two ways. If the values are fixed, they can be implemented using an LC network that acts as a filter that scales and phases shifts the signal. A couple of simple networks to use are the Pi network, or the T network, Figure 94. In order to determine the values of the impedances for the circuit, it is useful to start with the S-parameters for the network. The transmission coefficient of the network, S21, is equal to the weighting coefficient that is to be enforced. In order to ensure that there are no reflections from the network

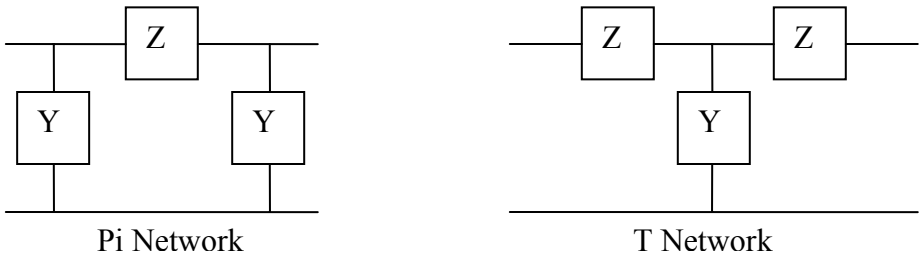


Figure 94. Simple lumped element networks .

that would return to the power combiners and possibly be distributed to other channels, the reflection coefficients, S11 and S22, of the network should be equal to zero and the intrinsic impedance of the network should be equal to 50 ohms. Finally, since the network is reciprocal, S12 will be equal to the conjugate of S21. Finding the network that satisfies these requirements,

$$\begin{matrix} S_{11} = 0 & S_{12} = w_{12} \\ S_{21} = w_{12}^{*} & S_{22} = 0 \end{matrix} \quad , \qquad [6.3]$$

will weight the signal from the nth coil for the mth receiver with a coefficient equal to wmn.

In most cases, however, the weighting coefficients will depend on the imaging application and will need to be variable. In this case, they can be implemented as a variable attenuator followed by a variable phase shifter, Figure 95. These devices are digitally controlled by the scanner and can be modified for different imaging protocols. If solid state devices are used, it is possible that the switching devices are fast enough to allow different weighting coefficients to be used for every plane in a multislice or multiplane sequence.

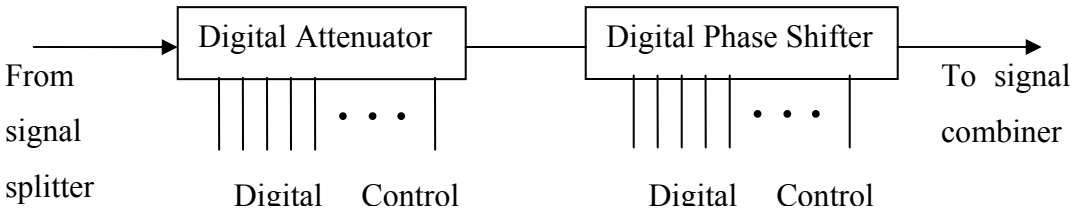


Figure 95. Digital control for implementing weighting coefficients.

The effectiveness of the combiner is determined by how well the weighting coefficients are synthesized. Using the lumped element networks, the coefficients can be implemented precisely, but are fixed and optimal for only one configuration. For the digitally controlled hardware combiner, the effectiveness is determined by the range and the resolution of the digital control. An example of digital and analog generated weighting coefficients for a 81 element grid array are shown in Figure 96. The resolution of the digital control is determined by smallest step size in the device. Using a digital attenuator the available resolution is up to .1 dB per step, but usually comes in 1dB minimum step size. The range of the attenuator is determined by the number of bits for control. A 4 bit attenuator with a resolution of 1dB has a range of 15dB. However, with .1dB resolution, the range is decreased by a factor of 10 to 1.5dB.

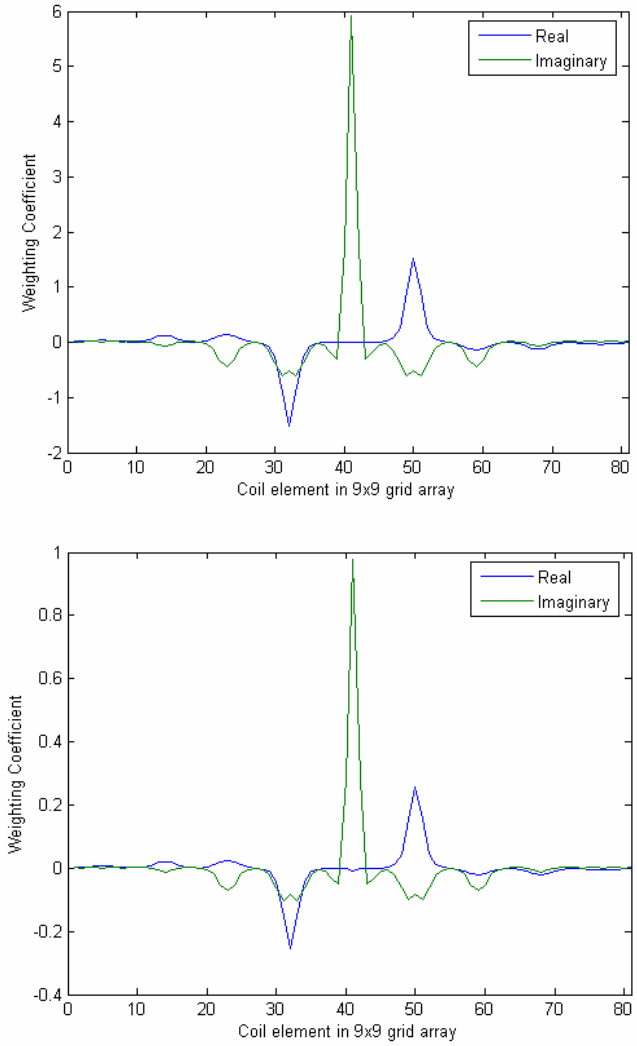


Figure 96. Analog and digital weighting coefficients.

The dynamic range and resolution of the weighting coefficients varies with imaging depth. Close to the array, the weights are more focused on a point and have greater variability. Farther array, the coefficients become more uniform. The dynamic range for a 9-by-9 grid array combined to one channel as the imaging depth is increase is shown in Figure 97. The array is combined to achieve optimal SNR at a point centered on the array. Near the surface of the array, the coefficients require the combiner to have approximately 40dB of range in the attenuators. As the distance is increased, the curve

generally decreases. At 4cm from the array, the coefficients have less than 10dB of dynamic range.

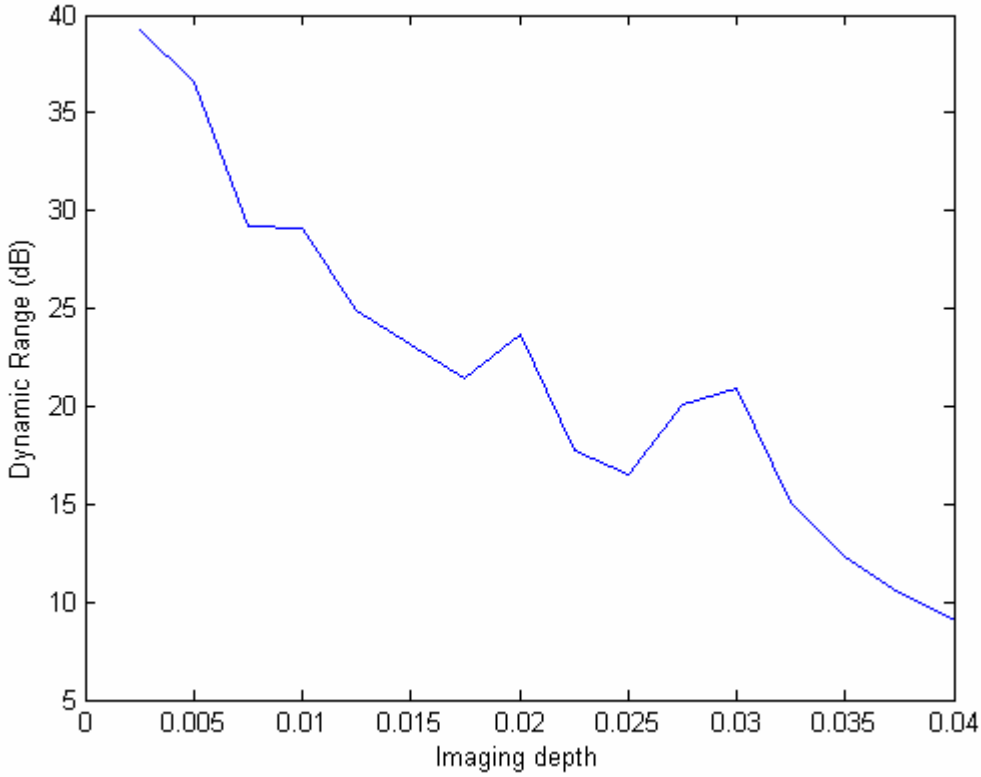


Figure 97. Coefficient dynamic range versus imaging depth.

The amount of RMS error in the weighting coefficients versus the amount of control bits were computed at various imaging depths, 2.5mm, 1cm, 2cm, and 3cm, and with resolutions of 1dB, 0.1dB, and 0.01dB. The number of bits of control for the attenuators and phase shifters were assumed to be the same. The resulting error curves are shown in Figure 98. As expected, the error decreases with higher resolution and greater control. Interestingly, as the imaging depth is increased the minimum error increases as the dynamic range decreases. This is due to the greater uniformity of the weighting coefficients that require greater resolution in the attenuators to approximate the values effectively.

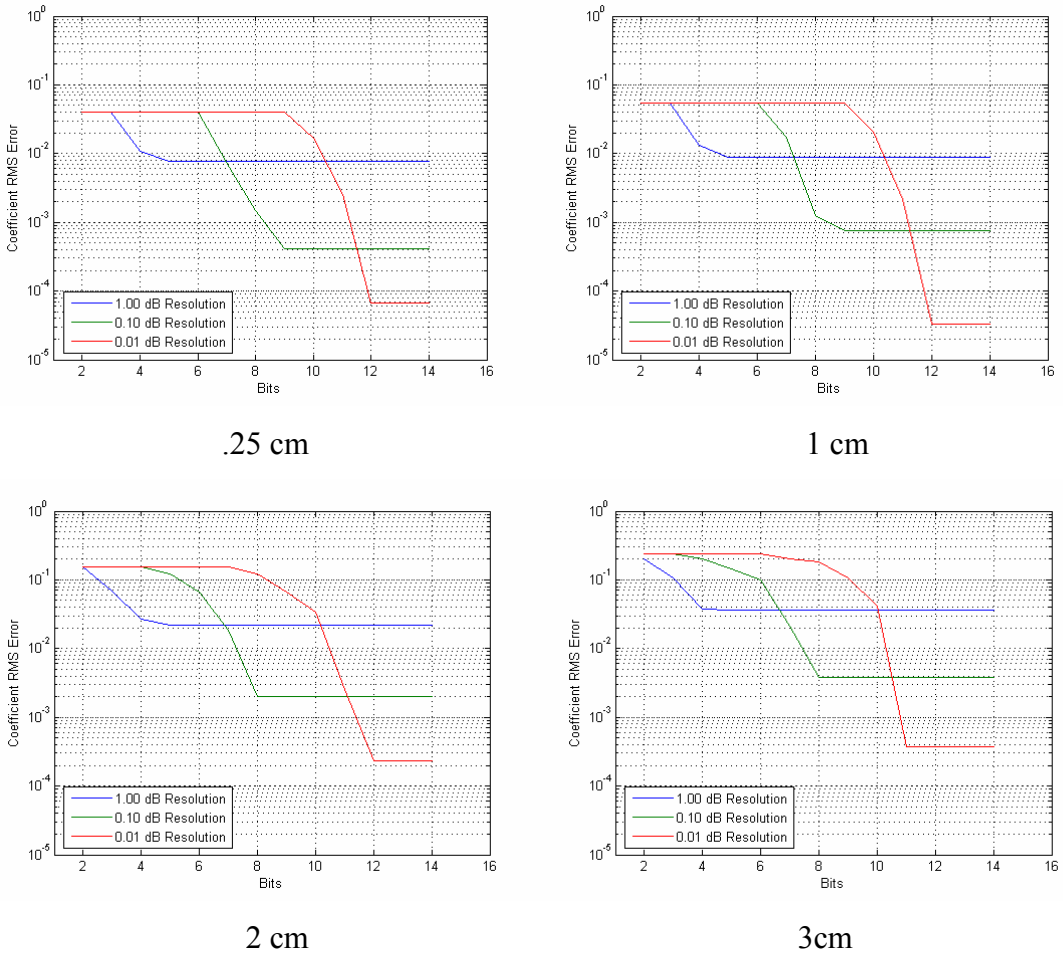


Figure 98. Error in weighting coefficients versus number of bits.

The RMS error for the image SNR maps using digitally constructed weighting coefficients was also computed for a 9x9 grid array combined to a single channel. These error curves are shown below in Figure 99. Generally, the error is less than 2% when using a combiner with 0.1dB resolution and 10 bits of dynamic range. Even though the error in the weighting coefficients tends to increase with depth, shown previously in Figure 98, the RMS error in the SNR maps improves as the distance from the array is increased. This shows that the process of signal combination becomes more tolerant of error as the distance from the array is increased.

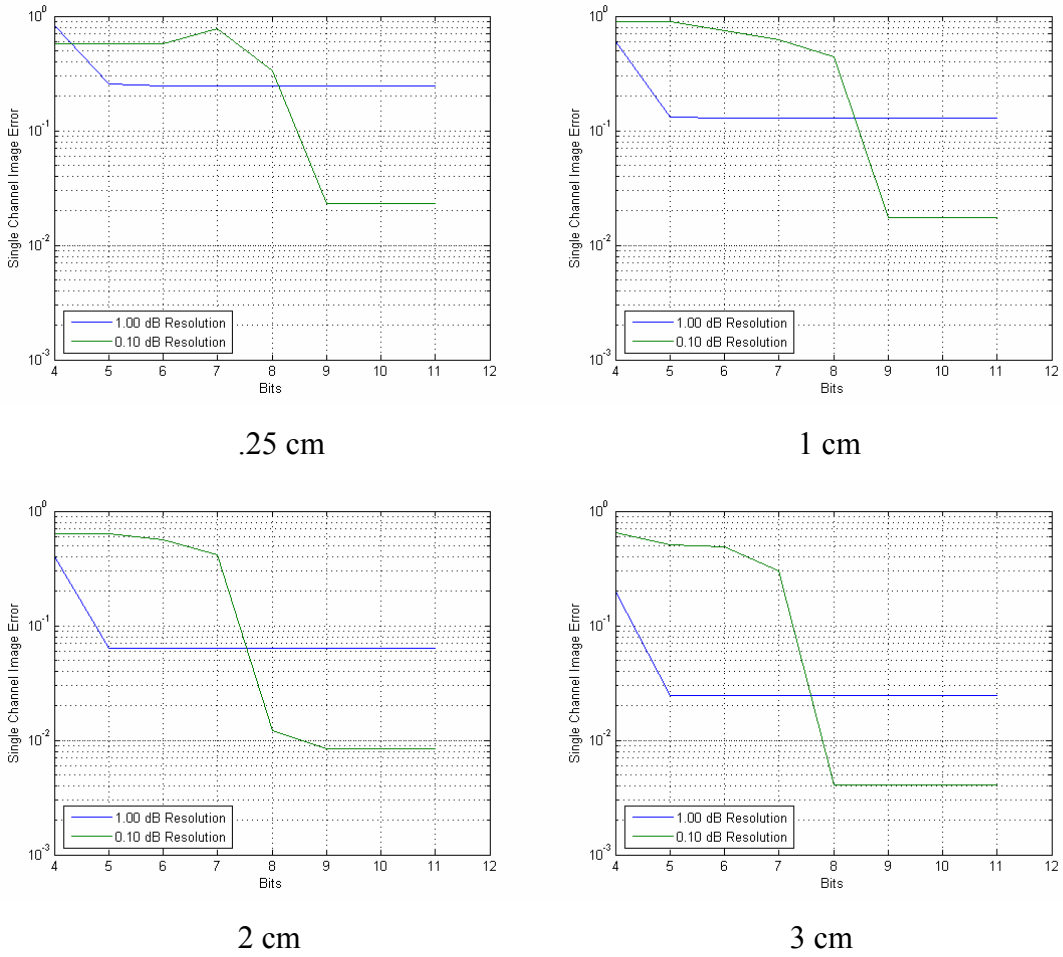


Figure 99. One channel image error from digital combiner.

CHAPTER VII

CONCLUSIONS

In this dissertation, tools for simulation and modeling arrays for MRI have been implemented and questions involving array combination for SNR optimization and SENSE reconstructions have been discussed.

- The optimal array design is a large array composed of simple well decoupled elements that allows the user the flexibility to effectively build a current sheet that maximizes the SNR at a point or over a region. The current sheet can be constructed either with software and a full bank of receiver channels to fully utilize the array, or can be done using hardware combiners to reduce the channel count to the minimum number of receivers necessary.
- The number of receiver channels required depends on the distance of the optimal point from the plane of the array. The current sheet that optimizes the SNR at this point can be related to an effective coil that is elliptical in shape and whose major axis dimension corresponds to the wire loop diameter that optimizes the SNR at that point. The number of receiver channels necessary is then just a function of the area of the array aperture divided by the area of the effective coil. Close to the array, many channels would be required. At a distance roughly equal to the length or width of the array, only one channel may be necessary.
- When only a few channels are available, it may be necessary to optimize the SNR for each channel over a region instead of at a point. When the elements are combined to optimize SNR at a point, the resulting channel sensitivity pattern will be highly focused on this point, particularly with the point is close to the plane of the array. Choosing weighting coefficients that optimize SNR over a region will broaden the sensitivity of the channel so that nulls don't appear in the combined SNR map for the array.
- When combining a large array for use with SENSE imaging, care must be taken to maintain the overall SNR of the array while minimizing g-factor. It was

shown in the results section that it is possible to achieve exceptional g-factor results at high reduction factors while at the same time having unacceptably deep nulls in the combined sensitivity pattern. However, using region combined channels can generate good SNR coverage, but result in poor g-factors, even at low reduction factors. In general, it appears that point combined arrays result in better SENSE performance as long as the minimum number of channels is used to give complete coverage of the FOV. This tradeoff between SNR and g-factor combinations needs further exploration.

- Signal combination can be implemented either in software or hardware. Combining in software requires that you have a complete receiver channel for every element in the physical array. The data from the each element is then weighted and combined in software and is used in order to reduce processing times for image reconstruction (23). Hardware combination reduces the number of receiver channels and the overall cost of the scanner. Ideally, the hardware combiner will have the variability to be adjusted to yield optimal results for the desired imaging protocol. This would be done with variable attenuators and phase shifters that provide the necessary resolution and dynamic range to properly approximate the weighting coefficients. The closer the imaging region is to the array, the tighter the error tolerances for the hardware combiner become. For specific imaging applications, the combiner could be simplified to use lumped element filters to generate a predefined set of weighting coefficients.

REFERENCES

1. Cowan B. Nuclear Magnetic Resonance and Relaxation. Cambridge: Cambridge University Press; 1997.

2. Haacke EM, Brown RW, Thompson MR, Venkatesan R. Magnetic Resonance Imaging: Physical Principles and Sequence Design. New York: John Wiley & Sons; 1999.

3. Haacke EM, Patrick JL. Reducing motion artifacts in two-dimensional Fourier transform imaging. Magn Reson Imaging 1986;4:359.

4. Wood ML, Henkelmen RM. MR image artifacts from periodic motion. Med Phys 1985;12:143.

5. Carlson JW. An algorithm for NMR imaging reconstruction based on multiple RF receiver coils. J of Mag Reson 1987;74:376-380.

6. Roemer PB, Edelstein WA, Hayes CE, Souza SP, Mueller OM. The NMR phased array. Magn Reson Med 1990;16(2):192-225.

7. Kwiat D, Einav S, Navon G. A decoupled coil detector array for fast image acquisition in magnetic resonance imaging. Med Phys 1991;18(2):251-265.

8. Pruessmann KP, Weiger M, Scheidegger MB, Boesiger P. SENSE: sensitivity encoding for fast MRI. Magn Reson Med 1999;42(5):952-962.

9. de Zwart JA, Ledden PJ, Kellman P, van Gelderen P, Duyn JH. Design of a SENSE-optimized high-sensitivity MRI receive coil for brain imaging. Magn Reson Med 2002;47(6):1218-1227.

10. Weiger M, Pruessmann KP, Leussler C, Roschmann P, Boesiger P. Specific coil design for SENSE: a six-element cardiac array. Magn Reson Med 2001;45(3):495-504.

11. Weiger M, Pruessmann KP, Boesiger P. Cardiac real-time imaging using SENSE. SENSitivity Encoding scheme. Magn Reson Med 2000;43(2):177-184.

12. Goldfarb JW, Holland AE. Parallel magnetic resonance imaging using coils with localized sensitivities. Magn Reson Imaging 2004;22(7):1025-1029.

13. Spence DK, Fujita H. A New Coil Array for SENSE Imaging with Four or Eight Receivers. In: Proc 10th Annual Meeting ISMRM, Honolulu, 2002. p 902.

14. Fujita H, Spence DK. A Novel 8-Channel "Saddle-Train" Array Coil for Cardiac SENSE Imaging at 1.5T. In: Proc 10th Annual Meeting ISMRM, Honolulu, 2002. p 858.

15. Fujita H, Spence DK. A Novel 8-Channel "Saddle-Train" Array Coil for Abdominal SENSE Imaging at 1.5T. In: Proc 10th Annual Meeting ISMRM, Honolulu, 2002. p 326.

16. Bollenbeck J, Vester M, Oppelt R, Kroeckel H, Schnell W. A High Performance Multi-channel RF Receiver for Magnetic Resonance Imaging Systems. In: Proc 13th Annual Meeting ISMRM, Miami, 2005. p 860.

17. KannengieBer SAR. SIEMENS Parallel Imaging Vendor Update. In: Proc 2nd Internation Workshop for Parallel Imaging, Zurich, Switzerland, 2004. p 99-100.

18. Yallapragada N, Wright SM, McDougall MP. A Compact 64 Channel Real-Time MRI Reconstruction System. In: Proc 13th Annual Meeting ISMRM, Miami, 2005. p 862.

19. Wiggins GC, Potthast A, Triantafullou C, Lin F, Benner T, Wiggins CJ, Wald L. A 96-channel MRI System with 23- and 90-channel Phase Array Head Coils at 1.5 Tesla. In: Proc 13th Annual Meeting ISMRM, Miami, 2005. p 671.

20. Spence DK, Wright SM. Effects of element widths and FOV on SENSE imaging. In: Proc 2nd Internation Workshop for Parallel Imaging, Zurich, Switzerland, 2004. p 111.

21. Wright SM, McDougall MP, Brown DG. Single Echo Acquisition MR Imaging. In: Proc 11th Annual Meeting ISMRM, Toronto, 2003. p 23.

22. McDougall MP, Wright SM. 64-channel array coil for single echo acquisition magnetic resonance imaging. Magn Reson Med 2005;54(2):386-392.

23. Reykowski A. Reducing the Number of RF Channels in Parallel Imaging. In: Proc 2nd Internation Workshop for Parallel Imaging, Zurich, Switzerland, 2004. p 41.

24. Wright SM, Magin RL, Kelton JR. Arrays of mutually coupled receiver coils: theory and application. Magn Reson Med 1991;17(1):252-268.

25. Harrington RF. Time-Harmonic Electromagnetic Fields. New York: McGraw-Hill, Inc.; 1961.

26. Balanis CA. Antenna Theory: Analysis and Design. New York: John Wiley & Sons, Inc.; 1997.

27. Collin RE. Foundations for Microwave Engineering. New York: McGraw-Hill, Inc.; 1992.

28. Jackson JD. Classical Electrodynamics. New York: John Wiley & Sons, Inc.; 1999.

29. Chang K. Microwave Solid State Circuits and Applications. New York: John Wiley & Sons, Inc.; 1994.

30. Cho Z-H, Jones JP, Singh M. Foundations of Medical Imaging. New York: John Wiley & Sons, Inc.; 1993.

31. Jin J. Electromagnetic Analysis and Design in Magnetic Resonance Imaging. New York: CRC Press, LLC; 1999.

32. Levitt MH. Spin Dynamics. Chichester, UK: John Wiley & Sons, LTD; 2001.

33. Press WH, Vetterling WT, Teukolsky SA, Flannery BP. Numerical Recipes in C++. Cambridge: Cambridge University Press; 2002.

34. Weiger M, Pruessmann KP, Boesiger P. 2D SENSE for faster 3D MRI. Magma 2002;14(1):10-19.

35. Dydak U, Weiger M, Pruessmann KP, Meier D, Boesiger P. SENSitivity-Encoded spectroscopic imaging. Magn Reson Med 2001;46(4):713-722.

36. Reykowski A, Wright SM, Porter JR. Design of matching networks for low noise preamplifiers. Magn Reson Med 1995;33(6):848-852.

37. Lee RF, Giaquinto RO, Hardy CJ. Coupling and decoupling theory and its application to the MRI phased array. Magn Reson Med 2002;48(1):203-213.

38. Kocharian A, Rossman PJ, Hulshizer TC, Felmlee JP, Riederer SJ. Determination of appropriate RF blocking impedance for MRI surface coils and arrays. Magma 2000;10(2):80-83.

39. Pozar DM. Microwave Engineering. Reading, Massachusetts: Addison-Wesley Publishing Company; 1990.

40. Horn RA, Johnson CR. Matrix Analysis. Cambridge: Cambridge University Press; 1985.

41. Lauterbur PC. Image formation by induced local interactions. Examples employing magnetic resonance. Nature 1973;243:190.

42. Mehring M, WeberruB VA. Object Oriented Magnetic Resonance. San Diego: Academic Press; 2001.

43. Sodickson DK, Manning WJ. Simultaneous acquisition of spatial harmonics (SMASH): fast imaging with radiofrequency coil arrays. Magn Reson Med 1997;38(4):591-603.

44. Griswold MA, Jakob PM, Nittka M, Goldfarb JW, Haase A. Partially parallel imaging with localized sensitivities (PILS). Magn Reson Med 2000;44(4):602-609.

45. Griswold MA, Jakob PM, Heidemann RM, Nittka M, Jellus V, Wang J, Kiefer B, Haase A. Generalized autocalibrating partially parallel acquisitions (GRAPPA). Magn Reson Med 2002;47(6):1202-1210.

46. Glover GH, Hayes CE, Pelc NJ, Edelstein WA, Mueller OM, Hart HR, Hardy CJ, O'Donnell M, Barber WD. Comparison of linear and circular polarization for magnetic resonance imaging. J of Mag Reson 1985;64:255-270.

47. Wright SM, Wald LL. Theory and application of array coils in MR spectroscopy. NMR Biomed 1997;10(8):394-410.

VITA

Dan Kenrick Spence was born in 1972 in Rolla, MO and grew up in San Antonio, TX. He received a B.S. in electrical engineering from Texas A&M University in 1995. In 1999, he received his M.S. in electrical engineering from the Magnetic Resonance Systems Lab (MRSL), Texas A&M. Since then, he has been working towards his Ph.D. in the MRSL. In the summer and fall of 2001 and summer of 2002, he worked at an internship at USA, Instruments, Inc., in Aurora, OH. There, he worked to design and build prototype coil arrays for parallel imaging applications. Two of these designs currently have patents pending. In 2004, he worked with Dupont on techniques using an array of superconducting coils and Nuclear Quadrupole Resonance (NQR) for applications in explosives detection. His present research pertains to how to best use large coil arrays for MRI and parallel imaging. He has accepted a job with GE Healthcare as an advanced RF coils engineer and currently resides in Brookfield, Wisconsin. He can be reached at Dan.Spence@ge.com or at

GE Healthcare
3200 N. Grandview Blvd., W-832
Waukesha, WI 53188.